

**PENGEMBANGAN APLIKASI *ANDROID* KAMUS *COMMAND LINE* (FYComm)
SEBAGAI MEDIA BANTU BELAJAR SISWA SMK NEGERI 1 BANTUL
KOMPETENSI KEAHLIAN TEKNIK KOMPUTER DAN JARINGAN**

TUGAS AKHIR SKRIPSI

Diajukan kepada Fakultas Teknik Universitas Negeri Yogyakarta
untuk Memenuhi Sebagian Persyaratan Guna Memperoleh Gelar Sarjana Pendidikan



Oleh:
Afied Hutomo
NIM 10520241008

**PROGRAM STUDI PENDIDIKAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA
2014**

**PENGEMBANGAN APLIKASI *ANDROID* KAMUS *COMMAND LINE* (FYComm)
SEBAGAI MEDIA BANTU BELAJAR SISWA SMK NEGERI 1 BANTUL
KOMPETENSI KEAHLIAN TEKNIK KOMPUTER DAN JARINGAN**

Oleh :

Afied Hutomo

NIM 10520241008

ABSTRAK

Penelitian ini bertujuan untuk mengembangkan perangkat lunak bernama *FYComm*. *FYComm* merupakan aplikasi kamus *command line* berbasis *android* sebagai media bantu belajar siswa SMK N 1 Bantul kompetensi keahlian Teknik Komputer dan Jaringan. Penelitian ini fokus pada proses perancangan sistem dan pengujian unjuk kerja sistem perangkat lunak pada aspek *functionality*, *efficiency*, *usability* dan *portability*. Perangkat lunak hasil dari penelitian ini diharapkan dapat membantu siswa SMK N 1 Bantul kompetensi keahlian Teknik Komputer dan Jaringan untuk memahami materi pelajaran yang berkaitan dengan pengoperasian sistem operasi *Linux* berbasis *command line*.

Proses rekayasa perangkat lunak ini menggunakan *waterfall process model* sebagai model proses pengembangan perangkat lunak. Tahapan pengujian pada penelitian ini dilakukan dalam rangka melakukan *verification and validation* (V&V) pada perangkat lunak. Proses *verification* perangkat lunak dilakukan dengan pengujian *white box testing* yang dilakukan dengan metode *basis path testing*. Proses *validation* dilakukan dengan *black box testing*, *alpha testing* dan uji materi.

Berdasarkan hasil penelitian pada perangkat lunak yang dikembangkan menunjukkan hasil : 1) perancangan Aplikasi *android* kamus *command line* (*Fycomm*) sebagai media bantu belajar siswa SMK N 1 Bantul Kompetensi keahlian Teknik Komputer dan Jaringan telah melalui serangkaian proses rekayasa perangkat lunak dari desain hingga pengujian telah sesuai dengan spesifikasi; 2) unjuk kerja dari Aplikasi *android* kamus *command line* sebagai media bantu belajar siswa SMK N 1 Bantul Kompetensi keahlian Teknik Komputer dan Jaringan secara keseluruhan masuk dalam kategori “sangat layak” dengan rincian hasil uji per faktor : *functionality* 89.9%, *efficiency* 91.11%, *usability* 86.67% dan *portability* 100%.

Kata Kunci : aplikasi, kamus, *command line*, *android*, *functionality*, *efficiency*, *usability*, *portability*.

LEMBAR PERSETUJUAN

Tugas Akhir Skripsi dengan Judul

**PENGEMBANGAN APLIKASI *ANDROID* KAMUS *COMMAND LINE* (*FYComm*)
SEBAGAI MEDIA BANTU BELAJAR SISWA SMK NEGERI 1 BANTUL
KOMPETENSI KEAHLIAN TEKNIK KOMPUTER DAN JARINGAN**

Disusun oleh:

Afied Hutomo

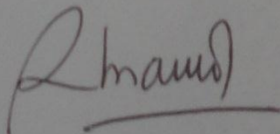
NIM. 10520241008

Telah memenuhi syarat dan disetujui oleh Dosen Pembimbing untuk dilaksanakan
Ujiak Akhir Tugas Akhir Skripsi bagi yang bersangkutan.

Yogyakarta, 14 Juli 2014

Mengetahui

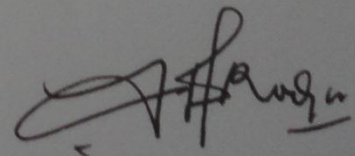
Ketua Program Studi
Pendidikan Teknik Informatika,



Dr. Ratna Wardani, S.Si., M.T
NIP. 19701218 200501 2 001

Disetujui,

Dosen Pembimbing,



Dr. Eko Marpanaji
NIP.19670608 199303 1 001

HALAMAN PENGESAHAN
TUGAS AKHIR SKRIPSI

PENGEMBANGAN APLIKASI ANDROID KAMUS COMMAND LINE (FYComm)
SEBAGAI MEDIA BANTU BELAJAR SISWA SMK NEGERI 1 BANTUL
KOMPETENSI KEAHLIAN TEKNIK KOMPUTER DAN JARINGAN

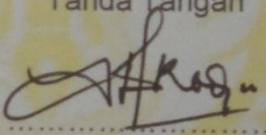
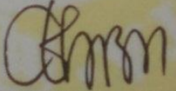
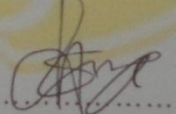
Disusun oleh:

Afied Hutomo

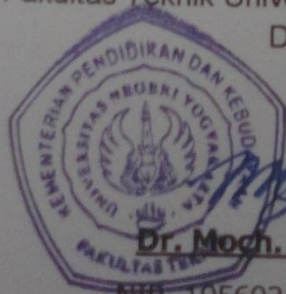
NIM. 10520241008

Telah dipertahankan di depan Tim Penguji Tugas Akhir Skripsi Program Studi
Pendidikan Teknik Informatika Fakultas Teknik Universitas Negeri Yogyakarta
pada tanggal 1 September 2014

TIM PENGUJI

| Nama/Jabatan | Tanda Tangan | Tanggal |
|--|--|----------------|
| <u>Dr. Eko Marpanaji</u> Ketua Penguji/Pembimbing |  | 17 / 09 / 2014 |
| <u>Athika Dwi Wiji Utami, M.Pd</u> Sekretaris |  | 15 / 09 / 2014 |
| <u>Nurkhamid, Ph.D</u> Penguji |  | 10 / 09 / 2014 |

Yogyakarta,
Fakultas Teknik Universitas Negeri Yogyakarta
Dekan,



Dr. Moch. Bruri Triyono

NIP. 19560216 198603 1 003 *te*

SURAT PERNYATAAN

Saya yang bertanda tangan dibawah ini:

Nama : Afied Hutomo

NIM : 10520241008

Program Studi : Pendidikan Teknik Informatika

Judul TAS : Pengembangan Aplikasi *Android* Kamus *Command Line*
(*FYComm*) sebagai Media Bantu Belajar Siswa SMK
Negeri 1 Bantul Kompetensi Keahlian Teknik Komputer
dan Jaringan.

Menyatakan bahwa skripsi ini benar-benar karya saya sendiri. Sepanjang sepengetahuan saya tidak terdapat karya atau pendapat yang ditulis atau diterbitkan orang lain kecuali sebagai acuan kutipan dengan mengikuti tata penulisan ilmiah yang telah lazim.

Yogyakarta, 15 Juli 2014

Yang menyatakan,



Afied Hutomo

NIM. 10520241008

HALAMAN MOTTO DAN PERSEMBAHAN

MOTTO

"Pendidikan adalah senjata paling mematikan, karena dengan itu Anda dapat mengubah dunia" .(Nelson Mandela).

"Seseorang yang berhenti belajar adalah orang lanjut usia, meskipun umurnya masih remaja. Seseorang yang tidak pernah berhenti belajar akan selamanya menjadi pemuda" (Henry Ford)

"Tugas kita bukanlah untuk berhasil. Tugas kita adalah untuk mencoba, karena didalam mencoba itulah kita menemukan dan belajar membangun kesempatan untuk berhasil" (Mario Teguh).

PERSEMBAHAN

Puji syukur kehadiran Allah SWT atas berkat rahmat dan karunia-Nya, sehingga Tugas Akhir Skripsi ini dapat terselesaikan.

Karya Skripsi ini aku persembahkan untuk Ayah dan Ibu, terimakasih atas segala doa, dukungan dan pengorbanan yang telah diberikan selama ini.

Hani adikku serta calon sabahat masa tuaku Erfin, terima kasih atas doa dan dukungannya.

Sahabat-sahabat terbaikku Bagus, Fera, Kusuma, Tami, dan Devy, kalian yang membuat duniaku berwarna dan menyenangkan.

dan teman-teman PTI-E 2010 yang selama empat tahun ini telah bersama menuntut ilmu dan memberikan masa-masa kebersamaan yang tak mungkin bisa terganti.

KATA PENGANTAR

Puji syukur kehadiran Allah SWT atas berkat rahmat dan karunia-Nya, Tugas Akhir Skripsi dalam rangka untuk memenuhi sebagian persyaratan untuk mendapatkan gelar Sarjana Pendidikan dengan Judul “Pengembangan Aplikasi *Android Kamus Command Line (FYComm)* sebagai Media Bantu Belajar Siswa SMK N 1 Bantul Kompetensi Keahlian Teknik Komputer dan Jaringan” dapat disusun sesuai dengan harapan. Tugas akhir skripsi ini dapat diselesaikan tidak lepas dari bantuan dan kerjasama dengan pihak lain. Berkenaan dengan hal tersebut, penulis menyampaikan ucapan terima kasih kepada yang terhormat

1. Bapak Dr. Eko Marpanaji selaku Dosen Pembimbing TAS yang telah banyak memberikan semangat, dorongan, dan bimbingan selama penyusunan Tugas Akhir Skripsi ini.
2. Ibu Dessy Irmawati, M.T dan Bapak Suparman, M.Pd selaku validator instrument penelitian TAS yang memberikan saran perbaikan sehingga penelitian TAS dapat terlaksana sesuai dengan tujuan.
3. Bapak Suparman, M.Pd dan Bapak Ponco Wali Pranoto, M.Pd selaku validator perangkat lunak yang memberikan masukan perbaikan pada perangkat lunak yang dikembangkan.
4. Bapak Muhammad Munir, M.Pd dan selaku Ketua Jurusan Pendidikan Teknik Elektronika dan Ketua yang telah membantu kelancaran selama proses penyusunan pra proposal sampai dengan selesai-nya TAS ini.
5. Ibu Dr. Ratna Wardani, S.Si., M.T selaku Ketua Program Studi Pendidikan Teknik Informatika yang telah membantu proses penyusunan pra proposal hingga selesainya TAS ini.

6. Bapak Dr. Moch. Bruri Triyono, M.Pd, selaku Dekan Fakultas Teknik Universitas Negeri Yogyakarta yang telah memberikan izin penelitian serta segala kemudahan yang diberikan.
7. Ibu Ir. Retno Yuniar Dwi Aryani selaku Kepala Sekolah SMK N 1 Bantul yang telah memberikan ijin dan bantuan dalam pelaksanaan penelitian Tugas Akhir Skripsi ini.
8. Ibu dan Bapak serta keluarga besar saya yang telah memberikan kasih sayang dan dukungan dari segala aspek sehingga tugas akhir skripsi ini dapat diselesaikan dengan baik.
9. Semua pihak, secara langsung maupun tidak langsung yang tidak dapat disebutkan disini atas bantuan dan perhatiannya selama penyusunan Tugas Akhir Skripsi ini.

Akhirnya, semoga segala bantuan yang telah diberikan semua pihak di atas menjadi amalan yang bermanfaat dan mendapatkan balasan dari Allah SWT dan Tugas Akhir Skripsi ini menjadi informasi yang bermanfaat bagi pembaca atau pihak lain yang membutuhkan.

Yogyakarta, Agustus 2014

Penulis

Afied Hutomo

NIM. 10520241008

DAFTAR ISI

| | |
|---|------|
| ABSTRAK..... | ii |
| LEMBAR PERSETUJUAN..... | iii |
| HALAMAN PENGESAHAN | iv |
| SURAT PERNYATAAN..... | v |
| HALAMAN MOTTO DAN PERSEMBAHAN | vi |
| KATA PENGANTAR | vii |
| DAFTAR ISI..... | ix |
| DAFTAR GAMBAR | xii |
| DAFTAR TABEL..... | xiv |
| DAFTAR LAMPIRAN | xvii |
| BAB I PENDAHULUAN | 1 |
| A. Latar Belakang..... | 1 |
| B. Identifikasi Masalah | 5 |
| C. Batasan Masalah | 5 |
| D. Rumusan Masalah..... | 6 |
| E. Tujuan Penelitian..... | 6 |
| F. Manfaat Penelitian | 7 |
| BAB II | 8 |
| KAJIAN PUSTAKA..... | 8 |
| A. Kajian Teori | 8 |
| 1. Kamus | 8 |
| 2. <i>Command Line Linux</i> | 10 |
| 3. Media Pembelajaran..... | 11 |
| 4. <i>Software Development Life Cycle</i> | 15 |
| 5. <i>Unified Modelling Language</i> | 22 |
| 6. <i>Software Testing</i> | 26 |
| 7. <i>Software Quality</i> | 32 |
| B. Kajian Penelitian yang relevan..... | 39 |
| C. Kerangka Pikir..... | 40 |

| | |
|---|-----|
| BAB III METODE PENELITIAN..... | 41 |
| A. Desain Penelitian | 41 |
| B. Prosedur Pengembangan | 43 |
| 1. Analisis Kebutuhan | 43 |
| 2. Desain | 45 |
| 3. Implementasi..... | 60 |
| 4. Pengujian..... | 65 |
| C. Variabel Penelitian..... | 66 |
| D. Tempat dan Waktu Penelitian..... | 67 |
| E. Subyek Penelitian..... | 67 |
| F. Metode Pengumpulan Data | 68 |
| 1. Observasi..... | 68 |
| 2. Kuesioner..... | 69 |
| G. Instrumen Penelitian | 69 |
| 1. Lembar Observasi..... | 70 |
| 2. Kuesioner..... | 72 |
| H. Teknik Analisis Data..... | 80 |
| BAB IV HASIL PENELITIAN DAN PEMBAHASAN | 82 |
| A. Hasil Penelitian | 82 |
| 1. Implementasi..... | 82 |
| 2. Pengujian Perangkat Lunak | 88 |
| B. Pembahasan Hasil Pengujian Pengujian Kualitas Perangkat Lunak..... | 112 |
| C. Hasil Akhir Produk..... | 114 |
| BAB V SIMPULAN DAN SARAN..... | 116 |
| A. Kesimpulan..... | 116 |
| B. Saran..... | 117 |
| DAFTAR PUSTAKA..... | 118 |
| LAMPIRAN | 120 |
| Lampiran 1 <i>Screenshot</i> Pengujian <i>Black Box</i> | 121 |
| Lampiran 2 <i>Screenshot</i> Pengujian Aspek <i>Portability</i> | 129 |
| Lampiran 3 Validasi Instrumen Penelitian | 144 |

| | |
|---|-----|
| Lampiran 4 Hasil Pengujian <i>Alpha</i> pada Oleh Ahli Media Pendidikan | 146 |
| Lampiran 5 Hasil Pengujian <i>Alpha</i> oleh <i>Independent Testing Group</i> | 151 |
| Lampiran 6 Hasil Pengujian Materi..... | 154 |
| Lampiran 7 Hasil Uji Reliabilitas Instrumen..... | 156 |
| Lampiran 8 Surat Ijin Penelitian | 158 |
| Lampiran 9 Surat Keterangan Telah Melakukan Penelitian | 161 |
| Lampiran 10 Surat Keterangan Pembimbing Skripsi..... | 162 |

DAFTAR GAMBAR

| | |
|---|----|
| Gambar 1. Simbol <i>Linux</i> | 10 |
| Gambar 2. Terminal pada Sistem Operasi <i>Linux</i> Ubuntu | 11 |
| Gambar 3. Pengembangan Model <i>Waterfall</i> | 14 |
| Gambar 4. Pengembangan Model <i>Prototype</i> | 16 |
| Gambar 5. Pengembangan Model <i>Spiral</i> | 17 |
| Gambar 6. <i>Flow Graph Notation</i> | 25 |
| Gambar 7. Macam-Macam Bentuk Loop(Perulangan) | 27 |
| Gambar 8. Rancangan <i>Use Case Diagram</i> | 50 |
| Gambar 9. <i>Sequence Diagram</i> Memulai Aplikasi | 51 |
| Gambar 10. <i>Sequence Diagram</i> Petunjuk | 52 |
| Gambar 11. <i>Sequence Diagram</i> Keluar Aplikasi | 52 |
| Gambar 12. <i>Sequence Diagram</i> Info Pengembang | 53 |
| Gambar 13. <i>Sequence Diagram</i> Cari Arti | 53 |
| Gambar 14. <i>Sequence Diagram</i> Cari Nama | 54 |
| Gambar 15. Diagram Alir Proses Pencarian | 55 |
| Gambar 16. Rancangan <i>Database</i> Tabel Kata Dasar dan Tabel Kamus | 56 |
| Gambar 17. Rancangan Halaman Utama | 56 |
| Gambar 18. Rancangan Halaman Menu Pencarian | 56 |
| Gambar 19. Rancangan Halaman Detail Informasi | 57 |
| Gambar 20. Tampilan Menu <i>Android SDK Manager</i> | 58 |
| Gambar 21. Tampilan Menu <i>Android Virtual Device Manager</i> | 59 |
| Gambar 22. Tampilan Menu <i>New</i> dalam <i>Android Virtual Device</i> | 60 |
| Gambar 23. <i>Interface IDE Eclipse</i> | 61 |

| | |
|--|-----|
| Gambar 24. Tampilan Aplikasi <i>SQLitebrowser</i> | 61 |
| Gambar 25. Hasil Rancangan Halaman Utama | 83 |
| Gambar 26. Hasil Rancangan Halaman Pencarian..... | 83 |
| Gambar 27. Hasil Rancangan Halaman Detail Informasi | 84 |
| Gambar 28. Hasil Rancangan Halaman Info Pengembang..... | 84 |
| Gambar 29. Hasil Rancangan Halaman Petunjuk..... | 85 |
| Gambar 30. Notasi Diagram Alir Proses Pencarian | 86 |
| Gambar 31. Diagram Persentase Hasil Pengujian <i>Alpha</i> Aspek <i>Functionality, Efficiency, Usability, dan Portability</i> | 111 |

DAFTAR TABEL

| | |
|---|----|
| Tabel 1. Simbol-Simbol pada <i>Use Case</i> | 21 |
| Tabel 2. Simbol-Simbol pada <i>Use Case</i> (lanjutan)..... | 22 |
| Tabel 3. Perbandingan Aspek <i>Software Quality</i> | 31 |
| Tabel 4. Definisi Aktor Aplikasi..... | 44 |
| Tabel 5. Definisi <i>Use Case</i> | 44 |
| Tabel 6. Definisi <i>Use Case</i> (lanjutan)..... | 45 |
| Tabel 7. Skenario <i>Use Case</i> Memulai Aplikasi..... | 46 |
| Tabel 8. Skenario <i>Use Case</i> Petunjuk | 46 |
| Tabel 9. Skenario <i>Use Case</i> Keluar Aplikasi..... | 47 |
| Tabel 10. Skenario <i>Use Case</i> Info Pengembang | 47 |
| Tabel 11. Skenario <i>Use Case</i> Cari Arti | 48 |
| Tabel 12. Skenario <i>Use Case</i> Cari Nama | 49 |
| Tabel 13. Skenario <i>Use Case</i> Cari Nama(lanjutan)..... | 50 |
| Tabel 14. Lembar Observasi Pengujian Aspek <i>Portability</i> | 68 |
| Tabel 15. Spesifikasi Uji Membuka Aplikasi | 69 |
| Tabel 16. Spesifikasi Uji Menu Utama | 70 |
| Tabel 17. Spesifikasi Uji Menu Pencarian | 70 |
| Tabel 18. Spesifikasi Uji Menu <i>Tab</i> Cari Arti..... | 70 |
| Tabel 19. Spesifikasi Uji Menu <i>Tab</i> Cari Arti(lanjutan) | 71 |
| Tabel 20. Spesifikasi Uji Menu <i>Tab</i> Cari Nama..... | 71 |
| Tabel 21. Spesifikasi Uji Menu <i>Tab</i> Cari Nama(lanjutan) | 72 |
| Tabel 22. Spesifikasi Uji Menu Petunjuk | 72 |
| Tabel 23. Spesifikasi Uji Menu Info Pengembang..... | 73 |

| | |
|--|-----|
| Tabel 24. Kisi-kisi Instrumen Pengujian <i>Functionality, Efficiency</i> , dan <i>Usability</i> | 74 |
| Tabel 25. Instrumen Pengujian <i>Functionality, Efficiency</i> , dan <i>Usability</i> | 75 |
| Tabel 26. Instrumen Pengujian <i>Functionality, Efficiency</i> , dan <i>Usability</i> (lanjutan) | 76 |
| Tabel 27. Interpretasi Persentase Likert | 78 |
| Tabel 28. Penyesuaian Interpretasi Persentase Likert | 78 |
| Tabel 29. Hasil Pengujian <i>Test Case</i> Aplikasi..... | 94 |
| Tabel 30. Hasil Pengujian Fungsionalitas Membuka Aplikasi..... | 95 |
| Tabel 31. Hasil Pengujian Fungsionalitas Menu Petunjuk..... | 96 |
| Tabel 32. Hasil Pengujian Fungsionalitas Keluar Aplikasi | 95 |
| Tabel 33. Hasil Pengujian Fungsionalitas Menu Info Pengembang..... | 97 |
| Tabel 34. Hasil Pengujian Fungsionalitas Menu Cari Arti..... | 97 |
| Tabel 35. Hasil Pengujian Fungsionalitas Menu Cari Arti(lanjutan) | 98 |
| Tabel 36. Hasil Pengujian Fungsionalitas Menu Cari Nama..... | 99 |
| Tabel 37. Hasil Pengujian Fungsionalitas Menu Cari Nama(lanjutan) | 100 |
| Tabel 38. Hasil Uji Spesifikasi Membuka Aplikasi | 101 |
| Tabel 39. Hasil Uji Spesifikasi Menu Utama..... | 101 |
| Tabel 40. Hasil Uji Spesifikasi Menu Pencarian | 102 |
| Tabel 41. Hasil Uji Spesifikasi Menu <i>Tab</i> Cari Arti | 103 |
| Tabel 42. Hasil Uji Spesifikasi Menu <i>Tab</i> Cari Nama | 104 |
| Tabel 43. Hasil Uji Spesifikasi Menu Petunjuk | 105 |
| Tabel 44. Hasil Uji Spesifikasi Menu Info Pengembang | 105 |
| Tabel 45. Hasil Instrumen Pengujian <i>Beta</i> Aspek <i>Functionality, Efficiency</i> , dan <i>Usability</i> | 107 |

| | |
|---|-----|
| Tabel 46. Hasil Pengujian Kualitas Perangkat Lunak Aspek <i>Portability</i> | 108 |
| Tabel 47. Perhitungan Hasil Pengujian <i>Beta</i> Aspek <i>Functionality</i> , <i>Efficiency</i> ,dan <i>Usability</i> | 110 |
| Tabel 48. Tingkat Kelayakan Perangkat Lunak dari Aspek <i>Functionality</i> , <i>Efficiency</i> , dan <i>Usability</i> | 111 |

DAFTAR LAMPIRAN

| | |
|---|-----|
| Lampiran 1 <i>Screenshot</i> Pengujian <i>Black Box</i> | 118 |
| Lampiran 2 <i>Screenshot</i> Pengujian Aspek <i>Portability</i> | 126 |
| Lampiran 3 Validasi Instrumen Penelitian | 140 |
| Lampiran 4 Hasil Pengujian <i>Alpha</i> oleh Ahli Media Pendidikan..... | 142 |
| Lampiran 5 Hasil Pengujian <i>Alpha</i> oleh <i>Independent Testing Group</i> | 147 |
| Lampiran 6 Hasil Pengujian Materi | 150 |
| Lampiran 7 Uji Reliabilitas Instrumen | 152 |
| Lampiran 8 Surat Ijin Penelitian | 154 |
| Lampiran 9 Keterangan Telah Melakukan Penelitian | 157 |
| Lampiran10 Pembimbing Skripsi | 158 |

BAB I

PENDAHULUAN

A. Latar Belakang

Sekolah Menengah Kejuruan atau SMK merupakan salah satu jenjang pendidikan yang dipersiapkan pemerintah sebagai lanjutan dari jenjang SMP/MTs. SMK lebih dipersiapkan untuk menghasilkan lulusan yang mampu menghadapi tuntutan dunia kerja. Hal ini sesuai dengan Undang-Undang Sistem Pendidikan Nasional (UUSPN) No. 20 tahun 2003 pasal 15, bahwa pendidikan menengah kejuruan bertujuan untuk menyiapkan peserta didik terutama untuk bekerja dalam bidang tertentu.

Kemampuan siswa lulusan SMK dalam menghadapi tuntutan dunia kerja ditentukan oleh kemampuan siswa tersebut dalam menguasai kompetensi kejuruan. Oleh karena itu, siswa wajib menguasai seluruh kompetensi kejuruan masing-masing jurusan. Semakin tinggi penguasaan siswa terhadap kompetensi kejuruan, semakin tinggi pula kemampuan dalam menghadapi tuntutan dunia kerja.

Salah satu kompetensi keahlian yang diselenggarakan di SMK N 1 Bantul adalah Teknik Komputer dan Jaringan (TKJ). Kompetensi keahlian Teknik Komputer dan Jaringan SMK N 1 Bantul mengharuskan siswanya menguasai beberapa kompetensi kejuruan. Kompetensi kejuruan tersebut diantaranya melakukan instalasi sistem operasi berbasis *command line*, membuat desain sistem keamanan jaringan, mengadministrasi server, dan merancang bangun *Wide Area Network* (WAN). Beberapa kompetensi keju-

ruan tersebut memiliki kesamaan yaitu menggunakan sistem operasi komputer Linux berbasis *command line*.

Pengoperasian sistem operasi komputer Linux berbasis *command line* (CLI) berbeda dengan pengoperasian sistem operasi komputer berbasis *Graphic User Interface* (GUI). Pengoperasian sistem operasi Linux berbasis *command line* membutuhkan keahlian tersendiri bagi penggunanya. Pengguna Linux berbasis *command line* mengubah konfigurasi sistem dengan cara mengetikkan perintah-perintah tertentu pada layar. Pengoperasian Linux berbasis *command line* dapat maksimal apabila pengguna mengetahui fungsi masing-masing perintah yang ada.

Beberapa siswa SMK N 1 Bantul Kompetensi keahlian Teknik Komputer dan Jaringan mengalami kendala dalam mengoperasikan sistem operasi komputer Linux berbasis CLI. Hal ini terlihat dari pengamatan langsung yang dilakukan oleh peneliti ketika melakukan kegiatan KKN PPL Tahun 2013 di SMK N 1 Bantul. Siswa sering menanyakan kepada peneliti tentang makna perintah *command line* tertentu. Hasil wawancara yang dilakukan peneliti kepada guru pengajar tentang kendala pembelajaran juga menyatakan hal yang sama. Beberapa siswa masih mengalami kendala dalam mengoperasikan sistem operasi Linux berbasis CLI. Siswa sekedar menyalin perintah yang ada di buku kemudian dipraktikkan tanpa mengetahui apa yang ia lakukan. Keadaan ini menyebabkan penyampaian materi kompetensi kejuruan oleh guru tidak dapat diterima siswa secara maksimal.

Permasalahan tersebut dapat terjadi karena beberapa faktor. Salah satunya karena sistem operasi komputer Linux masih dirasa asing bagi siswa. Menurut data dari *netmarketshare.com* (2014) hanya 1.48% user

menggunakan sistem operasi Linux diseluruh dunia. Masalah lain yaitu pemberian materi khusus dari guru tentang perintah-perintah *command line* belum ada sehingga siswa harus mencari sendiri tahu sendiri tentang arti perintah-perintah *command line*. Selain itu pengembangan sumber referensi untuk mengetahui arti dari perintah-perintah *command line* juga masih jarang ditemukan.

Salah satu permasalahan seperti yang sudah diungkapkan yaitu masih jarang ditemukannya sumber referensi untuk mengetahui arti dari perintah-perintah *command line* Linux. Permasalahan tersebut berusaha dipecahkan peneliti dengan cara mengembangkan sumber referensi alternatif. Salah satu alternatif pengembangan sumber referensi baru untuk perintah-perintah *command line* adalah dengan model kamus. Pemilihan sumber referensi alternatif tersebut didasarkan pada beberapa kelebihan yang dimiliki oleh kamus. Pemanfaatan media kamus sangat sederhana. Pengguna hanya tinggal mencari kata yang dicari sesuai abjad. Selain itu, informasi yang disajikan media kamus mudah dipahami. Alasan lain memilih model kamus yaitu terkait hasil beberapa penelitian yang sudah ada. Hasil dari beberapa penelitian yang sudah dilakukan menunjukkan bahwa media kamus terbukti efektif untuk meningkatkan hasil belajar siswa.

Sumber referensi untuk mengetahui arti perintah *command line* yang sudah ada masih berupa teks tertulis. Bentuk sumber referensi tersebut memerlukan usaha tersendiri bagi pengguna untuk memanfaatkannya. Pengguna harus mencari arti perintah tertentu secara manual. Sumber referensi lain yang sudah ada yaitu melalui menu *apropos*. Menu *apropos* sudah disediakan oleh sistem operasi Linux. Sumber referensi tersebut juga

tidak efisien karena pengguna harus menghidupkan komputer untuk memanfaatkannya. *Mobilitas* sumber referensi menjadi aspek yang penting. *Mobilitas* yang tinggi membuat sumber referensi dapat dimanfaatkan pengguna dimanapun dan kapanpun. Oleh karena itu sumber referensi yang baru akan dikembangkan pada *platform mobile*.

Menurut hasil riset dari GFK Asia bahwa pada periode Januari hingga September 2013 sebanyak 14,8 juta perangkat *smartphone* baru terjual di Indonesia. Keadaan tersebut menyebabkan *android* menguasai 72 persen pangsa pasar di Indonesia. Terkait dengan sasaran pengguna aplikasi nantinya yaitu siswa-siswi Kompetensi keahlian Teknik Komputer dan Jaringan SMK N 1 Bantul terdapat hasil riset yang dilakukan oleh Nielsen. Hasil riset tersebut adalah pada Juli 2012 sebanyak 58 persen remaja berusia 14 sampai 17 tahun telah memiliki *smartphone*. Beberapa hal inilah menjadi dasar pembuatan kamus *command line* di lingkungan *android*.

Berdasar uraian tersebut peneliti akan melakukan penelitian dengan judul **“Pengembangan Aplikasi *Android* Kamus *Command Line* (FYComm) sebagai Media Bantu Belajar Siswa SMK Negeri 1 Bantul Kompetensi Keahlian Teknik Komputer dan Jaringan”**. Penelitian ini akan menggunakan metode *Research and Development* (R&D). Penelitian yang dilakukan mencakup desain pengembangan sistem dan pengujian perangkat lunak aplikasi kamus *command line*. Diharapkan siswa kompetensi keahlian Teknik Komputer dan Jaringan SMK N 1 Bantul dapat lebih menguasai kompetensi kejuruan setelah menggunakan aplikasi *android command line* yang dibuat peneliti.

B. Identifikasi Masalah

Berdasarkan uraian latar belakang di atas, maka ada beberapa permasalahan yang dapat diidentifikasi yaitu:

1. Siswa SMK harus mampu menghadapi tuntutan dunia kerja.
2. Siswa SMK harus menguasai seluruh kompetensi kejuruan sesuai kompetensi keahlian masing-masing.
3. Siswa SMK N 1 Bantul kompetensi keahlian Teknik Komputer dan Jaringan mengalami kendala dalam mengoperasikan sistem operasi Linux berbasis *command line*.
4. Sistem operasi Linux belum banyak digunakan.
5. Pemberian materi tentang perintah-perintah *command line* oleh pengajar tidak ada.
6. Sumber referensi arti perintah-perintah *command line* yang sudah ada tidak efektif dan efisien.

C. Batasan Masalah

Berdasarkan identifikasi masalah yang ada, batasan masalah yang dibuat peneliti adalah sebagai berikut:

1. Masalah yang coba dipecahkan oleh peneliti terkait dengan sumber referensi arti perintah-perintah *command line* yang sudah ada dirasa tidak efektif dan efisien.
2. Penelitian ini sebatas menguji kelayakan perangkat lunak berdasarkan aspek kualitas perangkat lunak dari ISO 9126 yaitu *functionality*, *efficiency*, *usability*, dan *portability*

D. Rumusan Masalah

Berdasarkan batasan masalah dapat dirumuskan permasalahan sebagai berikut :

1. Bagaimanakah cara mengembangkan Aplikasi *Android* Kamus *Command Line* (*FYComm*) sebagai Media Bantu Belajar Siswa SMK Negeri 1 Bantul Kompetensi Keahlian Teknik Komputer dan Jaringan?
2. Bagaimanakah unjuk kerja Aplikasi *Android* Kamus *Command Line* (*FYComm*) sebagai Media Bantu Belajar Siswa SMK Negeri 1 Bantul Kompetensi Keahlian Teknik Komputer dan Jaringan berdasarkan aspek *functionality, efficiency, usability, dan portability*?

E. Tujuan Penelitian

Berdasarkan rumusan masalah di atas, tujuan dari penelitian ini adalah:

1. Mengetahui cara mengembangkan Aplikasi *Android* Kamus *Command Line* (*FYComm*) sebagai Media Bantu Belajar Siswa SMK Negeri 1 Bantul Kompetensi keahlian Teknik Komputer dan Jaringan.
2. Mengetahui unjuk kerja Aplikasi *Android* Kamus *Command Line* (*FYComm*) sebagai Media Bantu Belajar Siswa SMK Negeri 1 Bantul Kompetensi Keahlian Teknik Komputer dan Jaringan berdasarkan aspek *functionality, efficiency, usability, dan portability*.

F. Manfaat Penelitian

Manfaat yang diharapkan dari pengembangan Aplikasi Kamus *Command Line* Linux untuk *platform android* adalah

1. Manfaat Teoritis

- a. Membantu guru dalam menyampaikan materi kompetensi kejuruan khususnya yang berhubungan dengan sistem operasi Linux berbasis *command line*.
- b. Dapat dijadikan sebagai referensi pengembangan aplikasi pembelajaran berbasis *mobile*, khususnya untuk *platform android* bagi *developer*.

2. Manfaat Praktis

- c. Aplikasi Kamus *Command Line* untuk *platform android* ini dapat digunakan sebagai sumber referensi dalam pengoperasian sistem operasi Linux khususnya yang berbasis *Command Line*.
- d. Mendapatkan hasil uji kelayakan penggunaan aplikasi Kamus *Command Line* untuk *platform android* ditinjau dari aspek *software quality*.

BAB II

KAJIAN PUSTAKA

A. Kajian Teori

1. Kamus

Menurut Kamus Besar Bahasa Indonesia kamus memiliki beberapa pengertian. Pengertian yang pertama adalah buku acuan yang memuat kata dan ungkapan, biasanya disusun menurut abjad berikut keterangan tentang makna, pemakaian, dan terjemahannya. Pengertian yang kedua adalah buku yang memuat kumpulan istilah atau nama yang disusun menurut abjad beserta penjelasan tentang makna dan pemakaiannya.

Abdul Chaer (2007:179), menyebutkan pengertian kamus yang dikemukakan oleh beberapa para ahli yaitu

- a. Kridalaksana menyebutkan bahwa kamus adalah buku referensi yang memuat daftar kata atau gabungan kata dengan keterangan mengenai pelbagai segi maknanya dan penggunaannya dalam bahasa, biasanya disusun menurut abjad.
- b. Dalam *American Every Dictionary* disebutkan bahwa kamus adalah sebuah buku berisi kata-kata dari sebuah bahasa, biasanya disusun secara alfabetis, disertai keterangan akan artinya ucapannya, ejaannya, dan sebagainya.
- c. Labrousse menyebutkan bahwa kamus adalah buku berisi kumpulan kata-kata sebuah bahasa yang disusun secara alfabetis diikuti dengan definisi atau terjemahannya dalam bahasa lain.

- d. Keraf mendefinisikan kamus sebagai sebuah buku referensi, memuat daftar kata-kata yang terdapat dalam sebuah bahasa, disusun secara alfabetis, disertai keterangan cara menggunakan kata itu.

Secara umum kamus memiliki beberapa bagian yaitu

- a. Abjad,

Abjad adalah huruf awal kata.

- b. Lema atau entri atau kata dasar

Kata dasar adalah kata asli atau kata yang tidak diberi imbuhan apapun.

Contohnya : Merahasiakan, kata dasarnya adalah rahasia.

- c. Label

Label adalah penjelasan mengenai ragam bahasa, ragam ilmu kelas kata dan daerah atau negara asal dari kata yang dimaksud.

- d. Makna atau definisi

Makna adalah keterangan mengenai arti kata tersebut.

- e. Indeks

Indeks adalah kata yang terletak di sudut kanan atas untuk mempermudah pencarian.

Perangkat lunak yang akan dikembangkan oleh peneliti pada dasarnya menggunakan konsep seperti kamus. Perangkat lunak tersebut akan menampilkan informasi mengenai perintah-perintah yang ada pada *command line* Linux. Informasi perintah-perintah yang ditampilkan berupa pengertian, cara penggunaan, dan contoh penggunaannya.

2. *Command Line Linux*

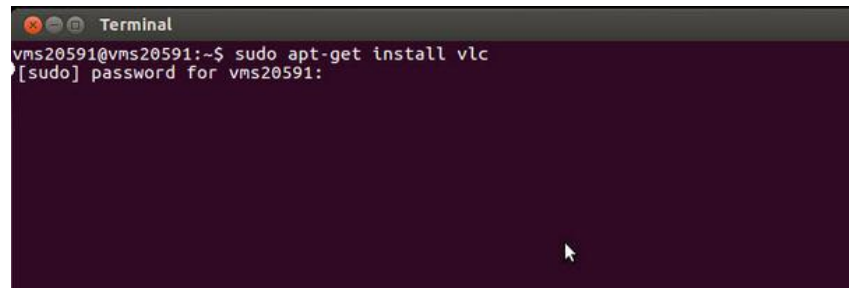
Sistem operasi Linux dibuat pertama kali oleh seorang mahasiswa Universitas of Helsinki di Finlandia yang bernama Linus Torvalds. Linux memiliki lisensi GNU, yaitu sebuah lisensi yang memungkinkan seseorang untuk mendistribusikan dan memodifikasi *source code* secara bebas dan gratis.



Gambar 1. Simbol Linux

Linux versi 0.01 dikerjakan bulan Agustus 1991. Kemudian pada tanggal 5 Oktober 1991, Linus Torvalds mengumumkan versi resmi Linux, yaitu versi 0.02 yang hanya dapat menjalankan *shell bash (GNU Bourne Again Shell)* dan *gcc (GNU C Compiler)*. Saat ini Linux adalah sistem UNIX yang sangat lengkap, bisa digunakan untuk jaringan, pengembangan *software* dan bahkan untuk pekerjaan sehari-hari.

Sebuah Sistem Operasi (OS) termasuk Linux, memerlukan suatu antar muka atau *interface* untuk *user* agar dapat menjalankan sistem operasi tersebut dengan mudah. Bentuk antar muka sistem operasi dapat dibagi menjadi dua yaitu *text based* dan *Graphical User Interface*. Sistem operasi Linux juga memiliki dua bentuk antar muka tersebut. Dalam perkembangannya bentuk antar muka *text based* atau *command line* lebih identik dengan sistem operasi Linux.



Gambar 2 . Terminal pada OS Linux Ubuntu

command line adalah tipe antarmuka dimana pengguna berinteraksi dengan sistem operasi melalui *text-terminal*. Pengguna menjalankan perintah dan program di sistem operasi tersebut dengan cara mengetikkan baris-baris tertentu. Meskipun konsepnya sama, tiap-tiap sistem operasi memiliki nama atau istilah yang berbeda untuk *command line* -nya.

Pengguna Linux mengenal *command line* pada Linux sebagai *terminal*. UNIX memberi nama *command line*-nya sebagai bash, ash, ksh, dan lain sebagainya. *Microsoft Disk Operating System* (MS-DOS) memberi nama *command.com* atau *Command Prompt*. Pada Windows Vista, Microsoft menamakannya *PowerShell*, sedangkan pada Apple namanya adalah *commandshell*.

3. Media Pembelajaran

Menurut Latuheru (1988:11), media adalah semua bentuk perantara yang digunakan oleh manusia untuk menyampaikan atau menyebar ide, sehingga hal yang dikemukakan itu bisa sampai pada penerima. Apabila media itu membawa pesan-pesan atau informasi yang bertujuan instruksional atau mengandung maksud-maksud pengajaran maka media itu disebut media pembelajaran. Manfaat media pembelajaran secara umum adalah untuk membantu siswa belajar secara optimal dan mempermudah interaksi pendidik

dengan siswa itu sendiri sehingga tujuan belajar tercapai. Menurut Harjanto (1997 : 245) seperti yang dikutip oleh Kusumah (2009), manfaat media pembelajaran antara lain memperjelas penyajian pesan agar tidak terlalu verbalistik, mengatasi keterbatasan ruang, waktu dan daya indera, serta dapat menimbulkan persepsi yang sama terhadap suatu masalah.

Terkait dengan hasil penelitian yang berbentuk aplikasi, menurut Supriyanto (2005 : 117) aplikasi adalah program yang memiliki aktivitas pemrosesan perintah yang diperlukan untuk melaksanakan permintaan pengguna dengan tujuan tertentu. Menurut Janner (2006 : 22) aplikasi adalah program atau sekelompok program yang dirancang untuk digunakan oleh pengguna akhir (*end user*). Aplikasi dapat dimanfaatkan untuk keperluan pembelajaran kepada siswa mengingat dalam suatu proses pembelajaran seharusnya terdapat interaksi antar komponen-komponen pembelajaran. Salah satu pendekatan pembelajaran yang memungkinkan antara komponen-komponen pembelajaran tersebut adalah pembelajaran interaktif.

Menurut Cahyadi. (2003), pembelajaran interaktif adalah lawan dari pembelajaran tradisional yaitu elemen yang disusun untuk meningkatkan pemahaman konsep secara interaktif dari siswa melalui kegiatan berpikir dan bekerja yang menghasilkan umpan balik melalui diskusi dengan petunjuk atau tanpa petunjuk dari pendidik (guru). Berdasarkan uraian di atas, dapat disimpulkan bahwa aplikasi pembelajaran interaktif adalah suatu program yang mengemas sebuah metode pembelajaran berbantuan komputer yang dapat memberikan respon balik terhadap pengguna akhir (siswa) dari apa yang telah diinputkan kepada aplikasi tersebut.

Menurut Romi Satrio.W (2006) penilaian terhadap aplikasi media pembelajaran dapat dikelompokkan menjadi tiga aspek yaitu aspek rekayasa perangkat lunak, aspek desain pembelajaran, dan aspek komunikasi. Masing-masing aspek tersebut memiliki sub karakteristik tertentu.

a. Aspek Rekayasa Perangkat Lunak

- 1) Efektif dan efisien dalam pengembangan maupun penggunaan media pembelajaran,
- 2) *Reliable* (mampu menampilkan fungsi-fungsi dengan tepat),
- 3) *Maintainable* (kemudahan dalam perbaikan jika terdapat kesalahan program),
- 4) Usabilitas (mudah digunakan dan sederhana dalam pengoperasiannya),
- 5) Ketepatan pemilihan jenis aplikasi/*software/tool* untuk pengembangan
- 6) Kompatibilitas (media pembelajaran dapat diinstalasi/dijalankan di berbagai *hardware* dan *software* yang ada),
- 7) Pemaketan program media pembelajaran terpadu dan mudah dalam eksekusi,
- 8) Dokumentasi program media pembelajaran yang lengkap meliputi: petunjuk instalasi (jelas, singkat, lengkap), *trouble shooting* (jelas, terstruktur, dan antisipatif), desain program (jelas, menggambarkan alur kerja program), dan
- 9) *Reusable* (sebagian atau seluruh program media pembelajaran dapat dimanfaatkan kembali untuk mengembangkan media pembelajaran lain).

b. Aspek Desain Pembelajaran

- 1) Kejelasan tujuan pembelajaran (rumusan, realistis),
- 2) Relevansi tujuan pembelajaran dengan SK/KD/Kurikulum,
- 3) Cakupan dan kedalaman tujuan pembelajaran,
- 4) Ketepatan penggunaan strategi pembelajaran,
- 5) Interaktivitas,
- 6) Pemberian motivasi belajar,
- 7) Kontekstualitas dan aktualitas,
- 8) Kelengkapan dan kualitas bahan bantuan belajar,
- 9) Kesesuaian materi dengan tujuan pembelajaran,
- 10) Kedalaman materi,
- 11) Kemudahan untuk dipahami,
- 12) Sistematis, runut, alur logika jelas,
- 13) Kejelasan uraian, pembahasan, contoh, simulasi, latihan,
- 14) Konsistensi evaluasi dengan tujuan pembelajaran,
- 15) Ketepatan dan ketetapan alat evaluasi, dan
- 16) Pemberian umpan balik terhadap hasil evaluasi.

c. Aspek Komunikasi Visual

- 1) Komunikatif yaitu sesuai dengan pesan dan dapat diterima/sejalan dengan keinginan sasaran,
- 2) Kreatif dalam ide berikut penuangan gagasan,
- 3) Sederhana dan memikat,
- 4) Audio (narasi, *sound effect*, *backsound*, musik),
- 5) Visual (*layout design*, *typography*, warna),
- 6) Media bergerak (animasi, *movie*), dan

7) *Layout Interactive* (ikon navigasi).

4. Software Development Life Cycle

Menurut Rosa dan Shalahuddin (2011:24), *Software Development Life Cycle* (SDLC) adalah proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya (berdasarkan *best practice* atau cara-cara yang sudah teruji dengan baik). SDLC berisi rencana-rencana rinci yang mengatur bagaimana untuk mengembangkan, memelihara, mengganti dan mengubah atau meningkatkan perangkat lunak tertentu.

Tahapan-tahapan pada SDLC secara umum yaitu:

a. Inisiasi

Inisiasi adalah tahap pertama dalam SDLC yang ditandai dengan pembuatan proposal proyek perangkat lunak.

b. Pengembangan Konsep Sistem

Pada tahap ini *developer* mendefinisikan lingkup konsep termasuk dokumen lingkup sistem, analisis manfaat biaya, manajemen rencana, dan pembelajaran kemudahan sistem.

c. Perencanaan

Tahap perencanaan dilakukan dengan mengembangkan rencana manajemen proyek dan dokumen perencanaan lainnya.

d. Analisis Kebutuhan.

Pada tahap ini *developer* akan menganalisis kebutuhan pemakai sistem perangkat lunak (*user*) dan mengembangkan kebutuhan *user*.

e. Desain

Tahap desain digunakan untuk mentransformasikan kebutuhan detail menjadi kebutuhan yang sudah lengkap. Tahap desain lebih fokus pada bagaimana dapat memenuhi fungsi-fungsi yang dibutuhkan.

f. Pengembangan

Tahap pengembangan digunakan untuk mengkonversi desain ke sistem yang lengkap. Tahap ini termasuk bagaimana memperoleh dan melakukan instalasi lingkungan sistem yang dibutuhkan.

g. Integrasi dan pengujian

Tahap ini digunakan untuk mendemonstrasikan sistem perangkat lunak bahwa telah memenuhi kebutuhan yang dispesifikasikan pada dokumen kebutuhan fungsional.

h. Implementasi

Tahap implementasi digunakan untuk melakukan penerapan perangkat lunak pada lingkungan produksi dan menjalankan resolusi dari permasalahan yang teridentifikasi pada tahap integrasi dan pengujian.

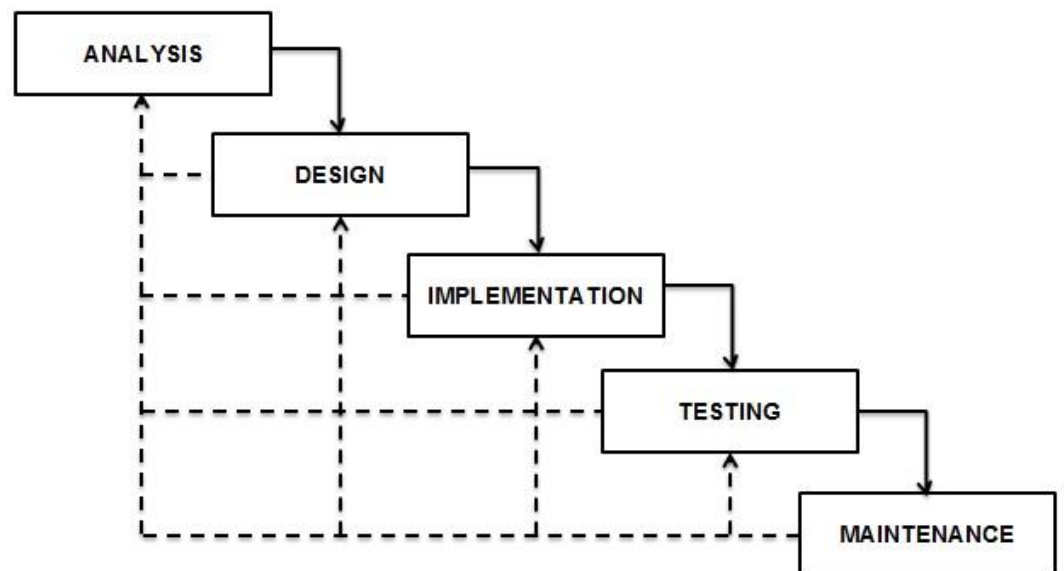
i. Pemeliharaan

Tahap pemeliharaan digunakan untuk mengoperasikan dan memelihara sistem pada lingkungan user.

Ada beberapa model SDLC yang dapat digunakan. Setiap model memiliki kelemahan dan kelebihan masing-masing. Hal yang terpenting adalah mengenali tipe user dan karakter pengembang. Beberapa model tersebut adalah *waterfall model*, *iterative model*, *spiral model*, dan *prototype model*.

a. *Waterfall Model*

Pengembangan model *waterfall* merupakan pendekatan SDLC yang pertama digunakan secara luas. Model *waterfall* membagi seluruh proses perangkat lunak menjadi fase yang terpisah. Hasil dari salah satu fase menjadi masukan untuk fase selanjutnya secara berurutan. Tahap *waterfall* dimulai dari *requirements analysis*, *design*, *coding*, dan *testing*.



Gambar 3. Pengembangan Model Waterfall

1) *Analysis*

Tahap ini merupakan proses pengumpulan kebutuhan yang dilakukan secara intensif. Tahap ini dilakukan untuk menyesuaikan antara kebutuhan perangkat lunak yang akan dikembangkan dengan kebutuhan *user*.

2) *Design*

Tahap desain dapat membantu dalam menentukan perangkat keras yang digunakan, persyaratan sistem, dan membantu dalam mendefinisikan arsitektur sistem secara keseluruhan. Hasil dari tahap desain digunakan untuk keperluan tahap implementasi atau pengkodean.

3) *Implementation*

Implementation merupakan tahap yang mengubah hasil tahap desain ke dalam bentuk program. Hasil tahap ini adalah perangkat lunak sesuai dengan desain yang telah dibuat pada tahap desain.

4) *Testing*

Tahap pengujian dilakukan untuk mengetahui kesalahan atau kegagalan perangkat lunak. Pengujian perangkat lunak fokus pada segi logika dan fungsionalitas.

5) *Maintenance*

Maintenance dilakukan untuk memperbaiki perangkat lunak yang dikembangkan berdasarkan hasil *testing*. Tahap ini dilakukan untuk meningkatkan kualitas perangkat lunak yang dikembangkan.

Kelebihan dari model *waterfall* adalah struktur tahap pengembangan yang dilakukan jelas. Selain itu pada setiap tahap pengembangan mode *waterfall* dihasilkan dokumentasi yang jelas pula. Kelebihan lain adalah tidak tumpang tindihnya antara tahap satu dengan yang lain.

Kelemahan dari model *waterfall* adalah tidak mampu untuk menghadapi perubahan kebutuhan selama pengembangan perangkat

lunak. Kelemahan lain adalah tidak adanya produk yang dihasilkan sampai tahap terakhir pada proses pengembangan. Hal ini mengakibatkan *user* sulit menjabarkan spesifikasi kebutuhan sistem secara lebih lanjut.

b. *Prototype Model*

Pengembangan model *prototype* dapat digunakan untuk menyambung ketidakpahaman antara pelanggan dengan pengembang perangkat lunak. Pelanggan dapat memperjelas spesifikasi kebutuhan yang diinginkan kepada pengembang perangkat lunak. Model *prototype* dimulai dengan mengumpulkan kebutuhan pelanggan terhadap perangkat lunak yang akan dibuat.

Tahap selanjutnya adalah membuat program *prototipe(mock-up)* agar pelanggan lebih terbayang dengan apa yang sebenarnya diinginkan. Program *prototype* tersebut dievaluasi oleh pelanggan sampai ditemukan spesifikasi yang sesuai dengan keinginan.



Gambar 4. Pengembangan Model Prototype

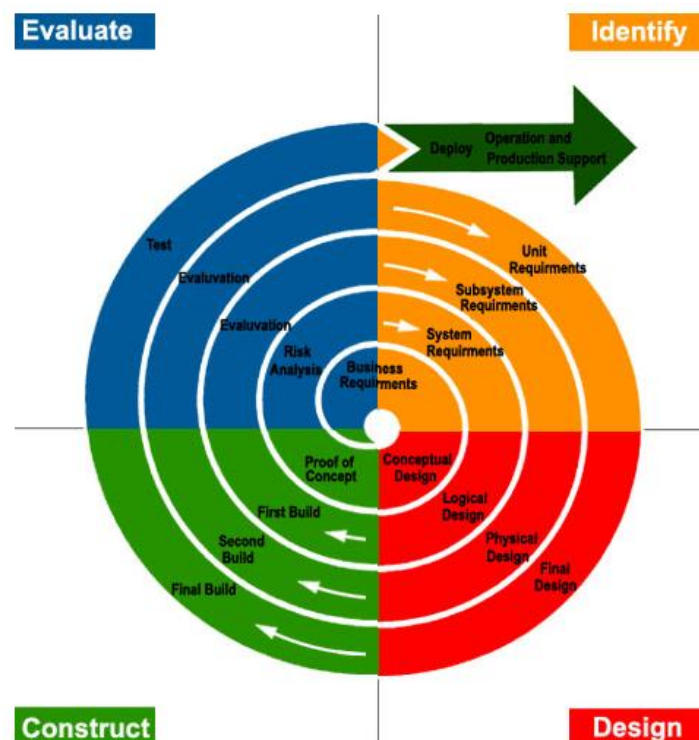
Model *prototype* digunakan untuk menjabarkan kebutuhan pelanggan secara detail. Hal ini dikarenakan pelanggan sering

kesulitan untuk menyampaikan kebutuhan secara detail. Kelemahan dari model *prototype* adalah pelanggan terlalu sering mengubah spesifikasi perangkat lunak. Hal ini dapat menghabiskan banyak tenaga serta waktu pengerjaan yang cenderung tidak tepat waktu.

c. *Spiral Model*

Pengembangan model *spiral* merupakan penggabungan ide pengembangan berulang (*prototyping*) dengan aspek sistematis terkendali model air terjun (*waterfall*). Model spiral meliputi manajemen resiko dalam pengembangan perangkat lunak.

Pengembangan model *spiral* dibagi menjadi beberapa *framework* aktivitas yang disebut dengan *task regions*. Aktivitas-aktivitas tersebut dibagi menjadi beberapa aktivitas. Berikut adalah aktivitas-aktivitas yang dilakukan dalam spiral model.



Gambar 5. Pengembangan Model Spiral

1) *Customer Communication*

Customer communication adalah aktivitas yang dibutuhkan untuk membangun komunikasi yang efektif antara *developer* dengan *user* terutama mengenai kebutuhan perangkat lunak.

2) *Planning*

Planning adalah aktivitas perencanaan yang dibutuhkan untuk menentukan sumberdaya, perkiraan waktu pengerjaan, dan informasi lainnya yang dibutuhkan untuk pengembangan software.

3) *Analysis Risk*

Analysis risk adalah aktivitas yang dijalankan untuk menganalisis baik resiko secara teknis maupun secara manajerial. Tahap inilah hanya dilakukan pada spiral model.

4) *Engineering*

Engineering adalah aktivitas yang dibutuhkan untuk membangun 1 atau lebih representasi dari aplikasi secara teknis.

5) *Construction & Release*

Construction and Release adalah aktivitas yang dibutuhkan untuk mengembangkan perangkat lunak, melakukan pengujian, instalasi dan penyediaan *user support* seperti *training* penggunaan *software* serta dokumentasi seperti buku manual penggunaan *software*.

6) *Customer Evaluation*

Customer evaluation adalah aktivitas yang dibutuhkan untuk mendapatkan *feedback* dari *user* berdasarkan evaluasi mereka

selama representasi *software* pada tahap engineering maupun pada implementasi.

Kelebihan pengembangan model *spiral* adalah mampu mengakomodasi perubahan kebutuhan yang dilakukan oleh *user*. Selain itu *user* dapat melihat sistem pada awal proses. Kelebihan lain adalah proses pengembangan dapat dibagi menjadi beberapa bagian kecil. Hal ini memungkinkan untuk mengembangkan lebih awal hal-hal yang lebih berisiko.

Kelemahan yang dimiliki oleh pengembangan model *spiral* adalah memerlukan proses manajemen yang lebih kompleks. Selain itu akhir dari proyek tidak dapat diketahui pada awal proses. Model pengembangan ini tidak cocok untuk proyek kecil. Hal ini dikarenakan dapat meningkatkan biaya yang dihabiskan untuk menyelesaikan proyek.

5. Unified Modelling Language

Pemodelan adalah proses merancang piranti lunak sebelum melakukan pengkodean. Model piranti dapat dianalogikan seperti membuat cetak biru pada pembangunan gedung. Membuat model dari sebuah sistem yang kompleks sangatlah penting karena kita tidak dapat memahami sistem semacam itu secara menyeluruh. Semakin kompleks sebuah sistem, semakin penting pula penggunaan teknik pemodelan yang baik.

Pada teknik perancangan berdasarkan objek *oriented*, hasil pemodelan konseptual digunakan untuk menggambarkan fungsionalitas sesuai persyaratan yang ditentukan pelanggan. Pemodelan konseptual itu

diwujudkan dalam bentuk sebuah standar untuk merancang model sebuah sistem yang bernama *Unified Modelling Language* (UML). Menurut Yuni Sugiarti(2013:34), UML adalah sebuah “bahasa” yang telah menjadi standar dalam industri untuk memvisualisasikan, merancang, dan mendokumentasikan sistem perangkat lunak.

Pada pemodelan *UML* 2.3 terdiri dari 13 macam diagram yang dikelompokkan dalam tiga kategori yaitu :

a. *Structure Diagrams*

Kumpulan diagram yang digunakan untuk menggambarkan suatu struktur status dari sistem yang dimodelkan.

b. *Behavior Diagrams*

Kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.

c. *Interaction Diagrams*

Kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun antar subsistem pada suatu sistem.

Tahap desain pada pengembangan perangkat lunak aplikasi android kamus *command line* sebagai media bantu belajar siswa SMK N 1 Bantul Kompetensi keahlian teknik Komputer dan Jaringan menggunakan *use case diagram* dan *sequence diagram*.

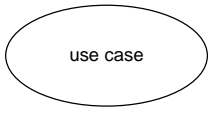

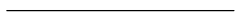
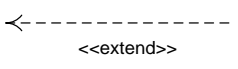
a. *Use case Diagram*

Use case atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar *use case* digunakan

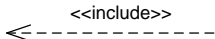
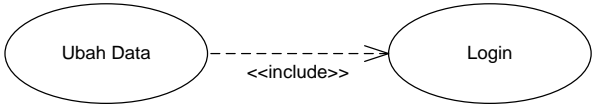
untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu (Rosa & Salahuddin, 2011:130).

Simbol-simbol yang ada pada *use case* ditunjukkan oleh tabel 1 dibawah ini

Tabel 1. Simbol-Simbol pada *Use Case*

| Simbol | Keterangan |
|---|--|
|  | <i>Use case</i> merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau actor. <i>Use case</i> dapat dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i> . |
|  | Aktor dapat berupa orang, proses atau sistem lain yang berinteraksi dengan sistem yang akan dibuat. Walaupun simbol dari aktor adalah gambar orang tapi aktor belum tentu merupakan orang. Aktor dapat dinyatakan menggunakan kata benda di awal frase nama aktor. |
|  | <i>Asosiation</i> menggambarkan komunikasi antar aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor. |
|  | <i>Extend</i> merupakan relasi tambahan antar <i>use case</i> . <i>Use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa use tambahan itu. Arah panah menunjuk pada <i>use case</i> yang dituju. |

Tabel 2. Simbol-Simbol pada *Use Case*(lanjutan)

| Simbol | Keterangan |
|--|---|
| | Berdasarkan gambar diatas dapat dikatakan bahwa Ubah Data sebagai <i>use case</i> utama dan Ubah Nama dan Ubah Alamat sebagai <i>use case</i> tambahan. <i>Use Case</i> Ubah Data meng- <i>extend</i> <i>use case</i> ubah Nama dan Ubah Alamat. Sehingga <i>use case</i> Ubah Data dapat dijalankan tanpa harus menjalankan <i>use case</i> Ubah Nama atau <i>use case</i> Ubah Alamat. |
|  | <p>Include merupakan relasi tambahan antar <i>use case</i>. <i>Use case</i> yang ditambahkan memerlukan <i>use case</i> utama untuk menjalankan fungsinya. Arah panah mengarah pada <i>use case</i> yang harus dijalankan terlebih dahulu.</p>  <p>Berdasarkan gambar diatas dapat dikatakan bahwa <i>use case</i> Login meng-include <i>use case</i> Ubah Data. Sehingga <i>use case</i> Ubah Data dapat dijalankan ketika aktor menjalankan <i>use case</i> Login terlebih dahulu.</p> |

b. *Sequence Diagram*

Diagram sekuen merupakan diagram yang menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antarobjek (Rosa & Salahuddin,2011:137). Banyaknya diagram *sequence* yang harus digambarkan adalah sebanyak pendefinisian *use case* yang memiliki proses sendiri. Penomoran pesan yang ada pada diagram *sequence* berdasarkan urutan interaksi pesan. Penggambaran letak pesan harus

berurutan dimana pesan yang lebih diatas dari lainnya adalah pesan yang berjalan terlebih dahulu. Diagram *sequence* memiliki ciri-ciri yaitu

- 1) Diagram *sequence* terdapat garis hidup objek. Garis hidup objek adalah garis tegak vertical yang mencerminkan eksistensi sebuah objek sepanjang periode waktu tertentu.
- 2) Pada diagram *sequence* terdapat *Focus of Control*. *Focus of Control* berupa persegi panjang ramping dan tinggi yang menampilkan aksi suatu objek secara langsung atau sepanjang sub ordinat.

6. Software Testing

Menurut Rosa dan Shalahuddin (2011:210), pengujian perangkat lunak atau *software testing* adalah satu set aktifitas yang direncanakan dan sistematis untuk menguji atau mengevaluasi kebenaran yang diinginkan. Pengujian terhadap perangkat lunak dilakukan untuk menghindari banyaknya kesalahan. Pengujian perangkat lunak tidak hanya dilakukan untuk meminimalisir kesalahan secara teknik tetapi juga kesalahan non teknis. Kesalahan non teknis tersebut seperti pengujian pesan kesalahan sehingga *user* tidak bingung.

Menurut Roger S. Pressman(2002:572), pengujian perangkat lunak merupakan salah satu elemen dari rekayasa perangkat lunak yang sering disebut dengan *verification and validation testing* V&V. Verifikasi sendiri mengacu pada serangkaian kegiatan yang memastikan perangkat lunak dapat melakukan fungsi-fungsi tertentu yang telah ditentukan. Validasi sendiri

mengacu pada suatu set yang memastikan perangkat lunak sudah sesuai dengan kebutuhan pengguna.

Teknik pengujian yang dapat dilakukan untuk menguji perangkat lunak diantaranya :

a. *Verification*

Tahapan uji verifikasi ini memastikan apakah pengembang telah mengembangkan perangkat lunak dengan spesifikasi dan cara yang benar. Pada tahapan *verification testing* ini dapat dilakukan dengan metode *white box testing* (Pressman, 2002:581).

Menurut Roger S. Pressman (2002:533). Pengujian *white box* adalah metode desain *test case* yang menggunakan struktur kontrol desain prosedural untuk memperoleh *test case*. Pengujian *white box* dapat memvalidasi struktur-struktur data yang penting.

Pengujian *white box* dapat membantu pengembang untuk mengoptimalkan kode program. Hal ini dilakukan dengan menghilangkan baris program yang dianggap dapat menyebabkan cacat tersembunyi. Kelemahan dalam pengujian *white box* adalah kadang-kadang sangat sulit meneliti setiap detail program untuk menemukan kesalahan yang tersembunyi.

Beberapa teknik yang dapat dilakukan untuk melakukan pengujian *white box* yaitu:

1) *Basis Path Testing*

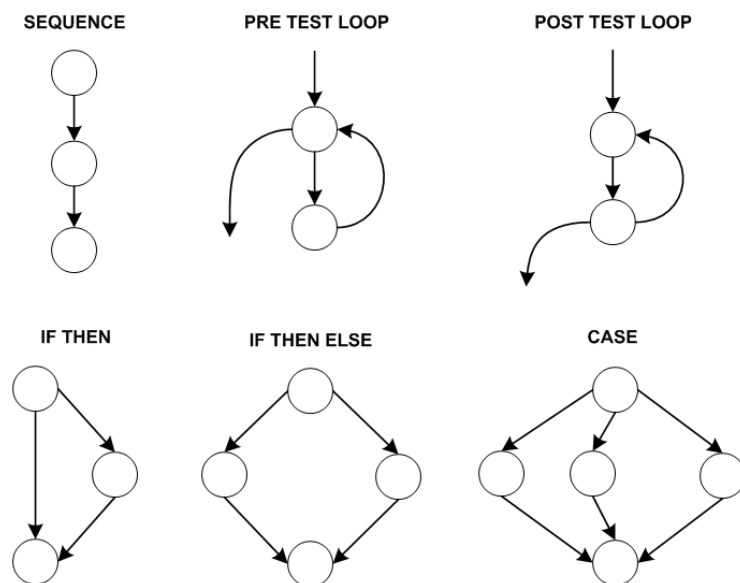
Metode *basis path testing* memungkinkan desainer *test case* mengukur kompleksitas logis dari desain prosedural. Ukuran

kompleksitas tersebut menjadi pedoman untuk menetapkan *basis set* dari jalur eksekusi.

Metode *basis path testing* dapat dilakukan dengan empat cara yaitu:

a) *Flow Graph Notation*

Flow Graph Notation merepresentasikan aliran kontrol logika yang digunakan untuk menggambarkan struktur kontrol program.



Gambar 6. Flow Graph Notation

b) *Cyclomatic Complexity*

Cyclomatic Complexity adalah metrik perangkat lunak yang memberikan pengukuran kuantitatif terhadap kompleksitas logis suatu program. *Cyclomatic Complexity* menentukan jumlah jalur independen dalam *basis set* suatu program.

c) *Graph Matrices*

Graph Matrices adalah matrik bujur sangkar yang ukurannya sama dengan jumlah simpul pada grafik alir. Masing-masing baris dan kolom sesuai dengan simpul yang telah diidentifikasi.

2) *Control Structure Testing*

Control Structure Testing merupakan teknik *white box testing* yang terdiri atas *conditional testing*, *data flow testing*, dan *loop testing*.

a) *Conditional Testing*

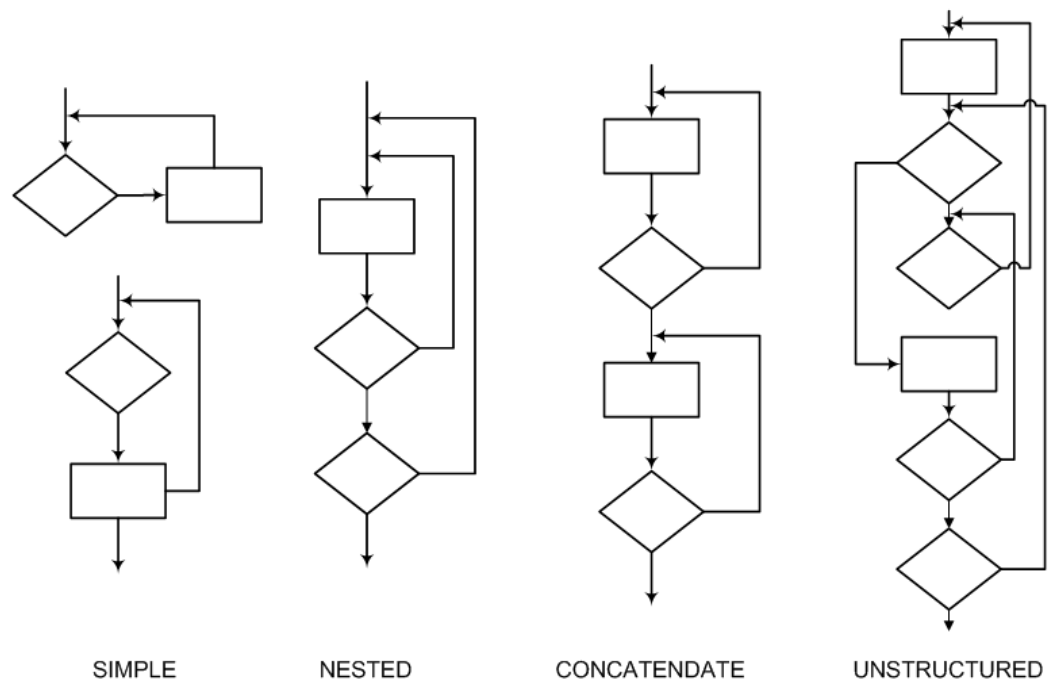
Conditional testing adalah sebuah metode desain *test case* yang menggunakan kondisi logis yang ada pada suatu program.

b) *Data Flow Testing*

Data flow testing digunakan untuk memilih jalur pengujian dari suatu program sesuai dengan lokasi definisi dan menggunakan variabel-variabel pada program.

c) *Loop Testing*

Loop testing merupakan teknik pengujian *white box* yang secara eksklusif berfokus pada validasi konstruksi loop. *Loop* dapat dibedakan menjadi empat kelas yaitu, loop sederhana (*simple*), loop terangkai (*nested*), loop tersarang (*concatendate*), dan loop tidak terstruktur (*unstructured*).



Gambar 7. Macam-Macam Bentuk Loop (Perulangan)

b. *Validation*

Validation testing merupakan pengujian yang dilakukan untuk menguji apakah perangkat lunak yang dikembangkan dapat melakukan apa yang benar-benar dibutuhkan pengguna secara fungsionalitas. *Validation testing* menjamin perangkat lunak yang dibuat telah sesuai dengan kebutuhan pengguna. Pada tahapan ini dapat dilakukan dengan pengujian *black box*, *alpha testing* dan *beta testing* (Pressman, 2002: 594-595).

1) *Black Box Testing*

Pengujian *Black Box* dilakukan untuk menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.

Pengujian *black box* dilakukan selama tahap akhir pengujian. Hal ini disebabkan karena pengujian *black box* memperhatikan struktur kontrol perangkat lunak.

Teknik dalam melakukan pengujian *black box* ada beberapa macam diantaranya:

a) *Equivalence Partitioning*

Menurut Roger S. Pressman (2002:556), *equivalence partitioning* adalah metode pengujian *black box* yang membagi domain input dari suatu program ke dalam kelas data. Teknik *equivalence partitioning* berusaha menentukan sebuah *test case* yang mengungkap kelas-kelas kesalahan. Hal ini akan mengurangi jumlah total *test case* yang harus dikembangkan.

b) *Boundary Value Analysis*

Menurut Roger S. Pressman (2002:558), *boundary value analysis* (BVA) adalah teknik desain proses yang melengkapi *equivalence partitioning*. Teknik *boundary value analysis* mengarah pada pemilihan *test case* pada “edge” dari kelas. Teknik *boundary value analysis* menyediakan beberapa pedoman dalam praktiknya. Pengaplikasian pedoman tersebut dapat mendeteksi kemungkinan kesalahan lebih besar.

c) *Comparison Testing*

Menurut Roger S. Pressman (2002:558), *comparison testing* dilakukan jika reliabilitas suatu perangkat lunak sangat kritis. *Comparison testing* dilakukan dengan mengembangkan perangkat lunak dalam versi independen yang berbeda. Setiap

versi perangkat lunak diuji dengan data yang sama untuk memastikan semua versi menghasilkan keluaran yang sama.

2) *Alpha Testing*

Pengujian *alpha* dilakukan oleh pengguna dalam organisasi pengembangan perangkat lunak. Hal ini dilakukan di lingkungan laboratorium sehingga tindakan pengguna dapat diukur dan dianalisis. Tujuannya adalah untuk mengukur kemampuan pengguna dalam menggunakan dan navigasi perangkat lunak sebelum dapat dirilis ke masyarakat umum. Pengujian alpha terdiri dari *unit testing*, *component testing*, dan *system testing*.

3) *Beta Testing*

Pengujian *beta* melibatkan sejumlah pengguna eksternal. Pengujian *beta* dilakukan untuk menguji program di situasi dunia nyata. Hal ini dilakukan untuk memastikan bahwa produk tersebut memiliki sedikit kesalahan atau *bug*. Pengujian *beta* membantu *developer* untuk mengkonfigurasi *customer resources and processes* yang akan dibutuhkan setelah peluncuran perangkat lunak.

7. **Software Quality**

Roger S.Pressman (2002:611) mendefinisikan *software quality* sebagai gabungan yang kompleks dari beberapa faktor yang akan bervariasi pada aplikasi dan pelanggan yang berbeda yang membutuhkannya. *Software quality* menentukan bagaimana suatu perangkat lunak memiliki suatu kualitas sehingga benar-benar berguna dalam pengaplikasiannya.

Sebuah perangkat lunak dikatakan berkualitas apabila memenuhi tiga ketentuan pokok yaitu :

- a. Perangkat lunak dapat memenuhi kebutuhan pemakai. Apabila sebuah perangkat lunak tidak dapat memenuhi kebutuhan pengguna, maka perangkat lunak tersebut dikatakan tidak atau kurang memiliki kualitas.
- b. Perangkat lunak memenuhi standar pengembangan software. Apabila sebuah perangkat lunak dikembangkan dengan tidak mengikuti metodologi standar, maka hampir dapat dipastikan bahwa kualitas yang baik akan sulit tercapai.
- c. Perangkat lunak memenuhi sejumlah kriteria implisit. Apabila sebuah perangkat lunak tidak memenuhi salah satu kriteria implisit tersebut, maka perangkat lunak tersebut tidak dapat dikatakan memiliki kualitas yang baik.

Kualitas perangkat lunak dapat dinilai melalui ukuran-ukuran, metode-metode, atau pengujian-pengujian *software* tertentu. Salah satu tolak ukur kualitas perangkat lunak adalah ISO 9126, yang dibuat oleh *International Organization for Standardization* (ISO) dan *International Electrotechnical Commission* (IEC). ISO 9126 adalah standar internasional yang diterbitkan oleh ISO untuk evaluasi kualitas perangkat.

Standar pengembangan menurut ISO 9126 dibagi menjadi empat bagian yang masing-masing menjelaskan model kualitas, metrik eksternal, metrik internal, dan metrik kualitas yang digunakan. Ada enam ukuran kualitas yang ditetapkan oleh ISO 9126, yaitu *functionality*, *reliability*, *usability*, *efficiency*, *portability*, dan *maintanability*.

Penelitian yang akan dilakukan oleh peneliti hanya akan menggunakan empat ukuran kualitas perangkat lunak yaitu *functionality*, *efficiency*, *usability*, dan *portability*. Pemilihan keempat aspek kualitas tersebut didasarkan pada analisis terhadap jurnal yang ditulis oleh Aida Niknejad yang berjudul “A Quality Evaluation of An Android Smartphone Application” dan Assaf ben David yang berjudul “Mobile Application Testing”.

Tabel 3. Perbandingan Aspek Software Quality

| Versi Aida Niknejad | Versi Assaf Ben David | Aspek yang Diambil dalam Penelitian |
|----------------------|-----------------------|-------------------------------------|
| <i>Functionality</i> | <i>Functionality</i> | <i>Functionality</i> |
| <i>Efficiency</i> | <i>Performance</i> | <i>Efficiency</i> |
| <i>Usability</i> | <i>Usability</i> | <i>Usability</i> |
| <i>Reliability</i> | <i>Compatibility</i> | <i>Portability</i> |

Menurut Niknejad (2011), kualitas perangkat lunak dapat diukur melalui empat aspek yaitu *functionality*, *reliability*, *usability*, dan *efficiency*. Aspek *reliability* berhubungan erat dengan koneksi jaringan, sehingga aspek tersebut tidak cocok dengan aplikasi yang dikembangkan oleh peneliti dimana bersifat *standalone*.

Menurut Assaf ben David, pengujian yang dilakukan untuk mengukur kualitas perangkat lunak berbasis *mobile* dapat dilakukan dengan empat cara yaitu *functionality testing*, *compatibility testing*, *usability testing*, dan *performance testing*. David menjelaskan bahwa *compatibility testing* berhubungan dengan beraneka ragamnya perangkat untuk mengakses perangkat lunak. Pengertian tersebut sama dengan pengertian aspek *portability* pada ISO 9126. Dengan demikian dapat dikatakan bahwa *compatibility testing* sama dengan pengujian aspek *portability*. Pengujian lain yaitu *performance testing* berhubungan dengan koneksi jaringan. Dengan

demikian *performance testing* tidak sesuai dengan penelitian yang dilakukan oleh peneliti.

Penjelasan-penjelasan diatas menjadi alasan peneliti untuk menggunakan empat aspek kualitas perangkat lunak yaitu *functionality*, *efficiency*, *usability*, dan *portability*. Pengertian dari masing-masing aspek kualitas perangkat lunak yang akan diujikan yaitu:

a. *Functionality*

Menurut ISO-9126 *functionality* adalah "*the capability of the software to provide functions which meet the stated and implied needs of users under the specified conditions of usage*". *Functionality* dapat menjadi tolak ukur bagaimana perangkat lunak yang dikembangkan dapat menjalankan fungsinya dan berperilaku sesuai dengan yang diharapkan.

Functionality memiliki lima sub karakteristik yaitu :

a. *Suitability*

Aspek *suitability* mengukur kemampuan perangkat lunak untuk menyediakan serangkaian fungsi yang sesuai untuk tugas – tugas tertentu dan tujuan pengguna.

b. *Accuracy*

Aspek *accuracy* mengukur kemampuan perangkat lunak dalam memberikan hasil yang presisi dan benar sesuai dengan kebutuhan.

c. *Security*

Aspek *security* mengukur kemampuan perangkat lunak untuk mencegah akses yang tidak diinginkan.

d. *Interopability*

Aspek *interopability* mengukur kemampuan perangkat lunak untuk berinteraksi dengan satu atau lebih sistem tertentu.

e. *Compliance*

Aspek *compliance* adalah aspek yang membuat perangkat lunak sesuai dengan aturan standar aplikasi.

Sub karakteristik aspek *functionality* yang digunakan dalam penelitian ini adalah *suitability*. Pemilihan sub karakteristik tersebut didasarkan pada jurnal yang ditulis oleh Aida Niknejad. *Suitability* penting karena pengembangan perangkat lunak *mobile* memiliki keterbatasan yang menyebabkan beberapa fungsi tidak dapat diakomodasi. Aspek *functionality* dianalisis melalui instrumen penelitian berupa kuesioner.

b. *Efficiency*

Menurut ISO 9126, *efficiency* merupakan kemampuan perangkat lunak untuk memberikan kinerja yang sesuai dan relatif terhadap jumlah sumber daya yang digunakan pada saat keadaan tersebut. *Efficiency* merupakan aspek yang berkaitan dengan respon, waktu pemrosesan, dan pemanfaatan sumber daya, yang mengacu pada sumber daya material (memory, cpu, koneksi jaringan) yang digunakan perangkat lunak.

Aspek *efficiency* memiliki dua sub karakteristik yaitu:

a. *Time Behavior*

Time behavior merupakan kemampuan perangkat lunak dalam memberikan respond dan waktu pengolahan yang sesuai saat melakukan fungsinya.

b. *Resource Behavior*

Resource behavior merupakan kemampuan perangkat lunak dalam menggunakan sumber daya yang dimilikinya ketika melakukan fungsi yang ditentukan.

Pada penelitian ini sub indikator *time behavior* dianggap sudah mewakili pengujian aspek *efficiency*. Hal tersebut berdasarkan pendapat yang dikemukakan oleh Aida Niknejad. Aspek *efficiency* dianalisis melalui instrumen penelitian berupa kuesioner.

c. ***Usability***

ISO 9126 mendefinisikan *usability* sebagai kemampuan perangkat lunak untuk dipahami, dipelajari, digunakan, dan menarik bagi pengguna ketika digunakan dalam kondisi tertentu. *Usability* merupakan kapabilitas perangkat lunak untuk dapat dimengerti, dipahami, dan digunakan (Agarwal,dkk,2010:91).

Menurut McCall dalam Pressman(2002:612) *usability* adalah usaha yang dibutuhkan untuk mempelajari, mengoperasikan, menyiapkan input, dan mengintreprestasikan *output* suatu program. Jadi *usability* dapat dikatakan sebagai sesuatu yang berkaitan dengan bagaimana sebuah produk dapat digunakan oleh pengguna untuk mencapai tujuan secara efektif, efisien dan memuaskan dari konteks penggunaan.

Menurut ISO 9126, aspek *usability* memiliki tiga *sub* karakteristik yaitu:

a. *Understandability*

Understandability merupakan kemampuan perangkat lunak dalam kemudahan untuk dipahami.

b. *Learnability*

Learnability merupakan kemampuan perangkat lunak dalam kemudahan untuk dipelajari.

c. *Operability*

Operability merupakan kemampuan perangkat lunak dalam kemudahan untuk dioperasikan.

Pengujian aspek usability menggunakan instrumen yang dibuat oleh peneliti berdasarkan indikator *sub* karakteristik. *Sub* karakteristik yang digunakan untuk aspek *usability* adalah *understandability*, *learnability*, dan *operability*.

d. ***Portability***

Menurut ISO 9126 *portability* adalah kemampuan perangkat lunak untuk ditransfer dari satu lingkungan ke lingkungan lain. Aspek *portability* berkaitan dengan usaha yang diperlukan untuk dapat mentransfer sebuah program dari sebuah lingkungan perangkat keras atau lunak tertentu ke lingkungan yang lain. Aspek *portability* memiliki empat sub karakteristik yaitu :

a. *Adaptability*

Adaptability merupakan kemampuan perangkat lunak untuk diadaptasikan pada lingkungan yang berbeda-beda.

b. *Instalability*

Instalability merupakan kemampuan perangkat lunak untuk diinstal dalam lingkungan yang berbeda-beda.

c. *Coexistence*

Coexistence merupakan kemampuan perangkat lunak untuk berdampingan dengan perangkat lunak lainnya dalam satu lingkungan dengan berbagi sumber daya.

d. *Replacability*

Replacability merupakan kemampuan perangkat lunak untuk digunakan sebagai pengganti perangkat lunak lainnya.

Pada penelitian ini indikator *installability* dan *adaptability* dianggap sudah mewakili pengujian dari aspek *portability*. Aspek *portability* pada platform *android* berkaitan dengan pengelompokan versi dari sistem operasi dan perbedaan besaran kedalaman piksel per inch yang dimiliki oleh setiap perangkat lunak berbasis android. Pengujian aspek portabilitas dilakukan dengan mengimplementasikan perangkat lunak ke beberapa versi dengan berbagai merek dan kedalaman piksel per inch yang berbeda.

B. Kajian Penelitian yang relevan

Beberapa hasil penelitian yang relevan dengan penelitian ini antara lain:

1. Penggunaan Media Pembelajaran Kamus Interaktif Bahasa Indonesia untuk Meningkatkan Motivasi dan Prestasi Siswa Kelas 7B UPTD SMPN 1 Ringinrejo Kabupaten Kediri Tahun pelajaran 2010/2011 (M. Faried.W, 2010). Tesis. Program Studi Teknologi Pendidikan Pascasarjana Universitas Sebelas Maret.

Hasil penelitian menunjukkan bahwa adanya peningkatan motivasi dan prestasi belajar siswa. Motivasi terlihat dari peningkatan presentasi skor

kuisisioner pada setiap kegiatan, sedangkan peningkatan prestasi terlihat pada peningkatan hasil belajar siswa pada setiap siklus.

2. Penggunaan Media Kamus dalam Pembelajaran Konstektual untuk Meningkatkan Kemampuan Siswa Memahami Makna Kata dalam Bacaan di Kelas VIID SMP Negeri 3 Gianyar(Ni Qayan Sri Damayanti, 2013).

Hasil dari penelitian ini adalah terdapat beberapa langkah pembelajaran yang tepat dengan menggunakan media kamus untuk meningkatkan kemampuan siswa memahami makna kata dalam bacaan. Peningkatan kemampuan siswa memahami makna kata dalam bacaan dengan menggunakan media kamus dalam pembelajaran kontekstual tergolong baik. Respon siswa terhadap penggunaan media kamus dalam pembelajaran memahami makna kata dalam bacaan tergolong sangat positif.

Hasil dari dua penelitian tersebut menunjukkan bahwa media pembelajaran kamus dapat membantu meningkatkan motivasi maupun prestasi siswa dalam belajar.

C. Kerangka Pikir

Aplikasi *FYComm* berbasis *android* ini merupakan aplikasi yang dikembangkan untuk membantu siswa kompetensi keahlian Teknik Komputer dan Jaringan SMK N 1 Bantul. Aplikasi ini diharapkan mampu membantu dalam mempelajari Kompetensi Kejuruan kompetensi keahlian Teknik Komputer dan Jaringan khususnya yang berkaitan dengan sistem operasi Linux berbasis *command line*.

Aplikasi *FYComm* berbasis *android* yang dikembangkan akan melalui proses *verification* dan *validation*. Proses *verification* akan dilakukan dengan

pengujian *white box*, sedangkan proses *validation* akan dilakukan dengan pengujian *black box* dan pengujian *alpha*. Selain dikaji berdasarkan empat pengujian tersebut, aplikasi juga akan dikaji berdasarkan materinya. Hasil dari proses *verification*, *validation* dan pengujian materi pada perangkat lunak yang dikembangkan akan dapat dijadikan acuan bahwa perangkat lunak ini termasuk perangkat lunak yang layak digunakan atau tidak.

BAB III

METODE PENELITIAN

A. Desain Penelitian

Metode yang digunakan dalam penelitian *Pengembangan Aplikasi Android Kamus Command Line sebagai Media Bantu Belajar Siswa SMK Negeri 1 Bantul Kompetensi keahlian Teknik Komputer dan Jaringan* ini adalah *Research and Development (R&D)*. Menurut Borg and Gall dalam Sugiyono (2007:9), *research and development* merupakan metode penelitian yang digunakan untuk mengembangkan atau memvalidasi produk-produk yang digunakan dalam pendidikan dan pembelajaran. *Research and Development* digunakan untuk menghasilkan produk tertentu dan menguji keefektifan produk tersebut(Sugiyono, 2007:407).

Penelitian yang akan dilakukan oleh peneliti berfokus pada proses pengembangan, verifikasi dan validasi perangkat lunak yang dikembangkan. Metode pengembangan perangkat lunak yang dilakukan peneliti mengacu pada model yang ada pada *Software Development Life Cycle (SDLC)*. Model SDLC yang digunakan dalam pengembangan penelitian ini adalah model *waterfall*. Beberapa tahap penelitian yang akan dilakukan dalam penelitian ini yaitu :

1. Analisis Kebutuhan

Analisis kebutuhan dilakukan untuk mengetahui kebutuhan yang dibutuhkan dalam proses pengembangan. Kebutuhan tersebut berupa fungsionalitas yang harus dimiliki perangkat lunak yang akan dikembangkan. Hasil dari analisis kebutuhan ini diharapkan dapat memenuhi semua kebutuhan yang diperlukan dalam proses pengembangan perangkat lunak.

2. Desain

Tahap desain dapat membantu dalam menentukan perangkat keras yang digunakan, persyaratan sistem. Tahap desain juga membantu dalam mendefinisikan arsitektur sistem secara keseluruhan. Proses desain akan berpedoman pada hasil dari analisis kebutuhan yang dilakukan. Desain yang dilakukan berbentuk rancangan antarmuka, rancangan basis data, rancangan diagram alir, dan *Unified Modelling Language (UML)*.

3. Implementasi

Implementation merupakan tahap yang mengubah hasil tahap desain ke dalam bentuk program. Implementasi hasil desain ke dalam bentuk program akan menggunakan bahasa pemrograman *java*.

4. Pengujian

Tahap pengujian berhubungan dengan proses verifikasi dan validasi (V&V) perangkat lunak. Proses verifikasi dilakukan dengan melakukan pengujian *white box* sedangkan proses validasi dilakukan dengan melakukan pengujian *black box*, pengujian *alpha*, dan pengujian *materi*.

B. Prosedur Pengembangan

Prosedur pengembangan perangkat lunak yang dilakukan peneliti berdasarkan *System Development Life Cycle Model Waterfall*. Prosedur tersebut terdiri dari analisis kebutuhan, desain, implementasi, dan pengujian.

1. Analisis Kebutuhan

Langkah pertama yang dilakukan dalam penelitian dan pengembangan ini adalah menetapkan ruang lingkup materi yang akan dikembangkan. Ruang lingkup materi yang akan dikembangkan yaitu bidang kompetensi keahlian SMK Kompetensi keahlian Teknik Komputer dan Jaringan. Langkah selanjutnya peneliti melakukan beberapa analisis dan kajian yang berkaitan dengan perangkat lunak yang akan dikembangkan. Proses analisis kebutuhan dilaksanakan dengan mencari dan mengkaji informasi mengenai perangkat yang akan dikembangkan baik secara langsung, maupun dari media cetak dan elektronik.

Analisis kebutuhan perangkat lunak meliputi observasi dan kajian literatur pendukung perancangan sistem perangkat lunak.

a) Observasi

Observasi dilakukan untuk mengetahui apa saja fitur yang diharapkan ada pada perangkat lunak yang akan dikembangkan. Fitur tersebut terkait dengan pengembangan aplikasi *android* kamus *command line*. Observasi dilakukan terhadap beberapa siswa dan guru pengajar kompetensi keahlian Teknik Komputer dan Jaringan di SMK N 1 Bantul.

Berdasarkan hasil observasi kelas dan wawancara dengan guru yang mengajar Kompetensi keahlian Teknik Komputer dan Jaringan diperoleh beberapa fungsi dalam konsep media yang akan dikembangkan. Beberapa fungsi yang dibutuhkan antara lain :

- 1) Sistem mampu menampilkan informasi mengenai perintah *command line* tertentu berdasarkan kata kunci yang berkaitan dengan nama perintah.
- 2) Sistem mampu menampilkan informasi mengenai perintah *command line* tertentu berdasarkan kata kunci yang berkaitan dengan keterangan perintah.
- 3) Sistem mampu menampilkan informasi mengenai nama, pengertian, dan contoh penggunaan perintah *command line* tertentu.

b) Studi Literatur

Studi literatur digunakan untuk mencari data mengenai konsep-konsep dan teori yang sudah ada untuk mendukung proses pengembangan perangkat lunak. Studi literatur dilakukan dengan melakukan

analisis terhadap penelitian-penelitian yang sudah dilakukan sebelumnya maupun terhadap aplikasi sejenis yang sudah ada.

Studi literatur yang telah dilakukan mendapatkan konsep teori mengenai beberapa hal, yaitu :

- 1) Media pembelajaran dengan media kamus mampu meningkatkan motivasi dan prestasi belajar.
- 2) Media pembelajaran dengan media kamus mampu meningkatkan kemampuan siswa dalam memahami makna kata.
- 3) *Software library* yang didukung dalam sistem operasi *android* adalah *Sqlite*.
- 4) *IDE Eclipse* dapat digunakan untuk mengembangkan aplikasi *android* dengan *plugin SDK Manager*.

2. Desain

a. Perancangan *Unified Modelling Language (UML)*

Desain perangkat lunak adalah proses multistep yang fokus pada desain pembuatan program. Fokus desain pada tahap ini termasuk didalamnya struktur data, arsitektur perangkat lunak, representasi antarmuka dan prosedur pengkodean(Rosa & Salahuddin,2011:27).

Desain pengembangan perangkat lunak yang dibuat nantinya akan menjadi panduan pengembang dalam proses pengkodean. Desain sistem perangkat lunak yang dibuat harus mempermudah pengembang dalam mengembangkan perangkat lunak. Desain model yang digunakan

nantinya harus sesuai dengan karakteristik perangkat lunak yang dikembangkan dan memperjelas pengembang.

Penelitian ini menggunakan desain sistem menurut metode *Unified Modelling Language (UML)*. Peneliti memilih model ini dikarenakan model tersebut sesuai dengan pengembangan sistem yang dilakukan yaitu berorientasi objek. UML yang dibuat peneliti yaitu *use case diagram* dan *sequence diagram*.

1) *Use Case Diagram*

Use case merupakan pemodelan untuk memodelkan kelakuan sistem yang dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih *actor* dengan sistem. *Use case* merupakan bentuk diagram yang menggambarkan fungsionalitas yang diharapkan dari sebuah sistem dilihat dari perspektif pengguna diluar sistem.

Use case juga dapat digunakan untuk merepresentasikan interaksi yang terjadi antara *actor* dengan proses sistem yang dibuat. Skenario pada perancangan *use case* merupakan langkah-langkah yang menerangkan urutan kejadian antara pengguna dengan sistem.

Tahap-tahap perancangan *use case* yang dilakukan peneliti yaitu :

a) Menentukan Definisi Aktor dan *Use Case* Aplikasi *FYComm*.

Berdasarkan hasil dari tahap analisis kebutuhan dapat diketahui tentang aktor dan apa yang diberikan sistem kepada aktor dalam hal ini *use case*. Penjelasan tentang aktor dan *use case* ditunjukkan pada tabel 4 dan tabel 5 berikut ini.

Tabel 4. Definisi Aktor *User*

| Aktor | Deskripsi |
|-------|-----------|
|-------|-----------|

| | |
|------|---|
| User | User merupakan aktor dari perangkat lunak <i>Find your Command</i> pada <i>platform</i> Android yang dapat melihat halaman petunjuk, halaman info pengembang dan memulai aplikasi untuk melakukan pencarian kata. |
|------|---|

Tabel 5. Definisi *Use case*

| Use Case | Deskripsi |
|------------------|---|
| Memulai Aplikasi | Memulai aplikasi merupakan proses dari perangkat lunak <i>Find Your Command</i> yang berfungsi untuk menampilkan halaman utama dan memulai inisiasi database kamus. |
| Petunjuk | Petunjuk merupakan proses dari perangkat lunak <i>Find Your Command</i> pada <i>platform</i> Android yang dapat menampilkan halaman petunjuk penggunaan aplikasi / perangkat lunak. |
| Keluar Aplikasi | Keluar Aplikasi merupakan proses dari perangkat lunak <i>Find Your Command</i> pada <i>platform</i> Android yang dapat digunakan untuk keluar dari aplikasi. |

Tabel 6. Definisi *Use case* (lanjutan)

| Use Case | Deskripsi |
|-----------------|---|
| Info Pengembang | Info Pengembang merupakan proses dari perangkat lunak <i>Find Your Command</i> pada <i>platform</i> Android yang dapat menampilkan halaman identitas pengembang aplikasi. |
| Cari Arti | Cari Arti merupakan <i>use case</i> dari perangkat lunak <i>Find your Command</i> pada <i>platform</i> Android yang dapat melakukan pencarian arti perintah CLI tertentu berdasarkan nama pada basisdata. |
| Cari Nama | Cari Nama merupakan proses dari perangkat lunak <i>Find your Command</i> pada <i>platform</i> Android yang dapat melakukan pencarian nama perintah CLI tertentu berdasarkan kata dasar tertentu pada |

| | |
|--|------------|
| | basisdata. |
|--|------------|

b) Skenario *Use Case* Aplikasi *FYComm*.

Berdasarkan *use case* tersebut maka dibuatlah skenario aksi aktor dan reaksi sistem. Skenario yang dibuat peneliti berdasarkan *use case* yaitu :

Nama Use Case : Memulai Aplikasi
Aktor : *User*
Deskripsi : Proses ini adalah kegiatan untuk membuka aplikasi Find Your Command.
Pre-Condition : -
Post-Condition : Menampilkan halaman menu utama aplikasi

Tabel 7. Skenario *Use Case* Memulai Aplikasi

| Aksi Aktor | Reaksi Sistem |
|---------------------|---|
| Alur Dasar | |
| 1. Memulai aplikasi | |
| | 2. Memanggil <i>file</i> HalamanUtama.java untuk menampilkan menu utama aplikasi. |
| | 3. Melakukan inisiasi hubungan ke <i>database</i> . |

Nama Use Case : Petunjuk
Aktor : *User*

Deskripsi : Proses ini adalah kegiatan untuk memunculkan petunjuk-petunjuk penggunaan melalui *screenshot* aplikasi.

Pre-Condition : User harus sudah berada di menu utama aplikasi.

Post-Condition : Petunjuk penggunaan aplikasi ditampilkan.

Tabel 8. Skenario *Use Case* Petunjuk

| Aksi Aktor | Reaksi Sistem |
|----------------------------|---|
| Alur Dasar | |
| 1. Menekan tombol Petunjuk | |
| | 2. Memanggil <i>file</i> HalamanPetunjuk.java untuk menampilkan gambar petunjuk pemakaian aplikasi. |

Nama Use Case : Keluar Aplikasi

Aktor : *User*

Deskripsi : Proses ini adalah kegiatan untuk keluar dari aplikasi.

Pre-Condition : *User* harus sudah berada di menu utama aplikasi.

Post-Condition : Menutup aplikasi Find Your Command.

Tabel 9. Skenario *Use Case* Keluar Aplikasi

| Aksi Aktor | Reaksi Sistem |
|--------------------------|--|
| Alur Dasar | |
| 1. Menekan tombol Keluar | |
| | 2. Memanggil <i>method</i> keluar untuk menghentikan aplikasi. |

Nama Use Case : Info Pengembang

Aktor : *User*

Deskripsi : Proses ini adalah kegiatan untuk memunculkan identitas pengembang aplikasi.

Pre-Condition : *User* harus sudah berada di menu utama aplikasi.

Post-Condition : Identitas pengembang aplikasi ditampilkan.

Tabel 10. Skenario *Use Case* Info Pengembang

| Aksi Aktor | Reaksi Sistem |
|---------------------------|--|
| Alur Dasar | |
| 1. Menekan tombol Tentang | |
| | 2. Memanggil <i>file</i> HalamanTentang untuk menampilkan identitas pengembang aplikasi. |

Nama Use Case : Cari Arti

Aktor : *User*

Deskripsi : Proses ini adalah kegiatan untuk mencari keterangan tentang *perintah command line* tertentu dalam basisdata berdasarkan input nama perintah *command line* tertentu.

Pre-Condition : *User* harus sudah berada di menu Pencarian

Post-Condition : Keterangan tentang perintah *command line* tertentu ditampilkan.

Tabel 11. Skenario *Use Case* Cari Arti

| Aksi Aktor | Reaksi Sistem |
|------------|---------------|
| Alur Dasar | |

| | |
|--|--|
| 1. Menekan menu tab Cari Arti | |
| | 2. Menampilkan menu tab Cari Arti |
| 3. Memasukkan nama perintah <i>command line</i> tertentu | |
| | 4. Menampilkan <i>autosuggest</i> nama perintah <i>command line</i> . |
| 5. Menekan salah satu pilihan yang ditampilkan <i>autosuggest</i> . | |
| 6. Menekan tombol Cari Arti | |
| | 7. Mengecek basisdata perintah <i>command line</i> tertentu berdasarkan input. |
| | 8. Jika data yang dimasukkan ditemukan dalam basis data, maka data akan ditampilkan |
| 1. Menekan pilihan data yang ditampilkan | |
| | 2. Menampilkan detail keterangan mengenai data perintah <i>command line</i> sesuai pilihan |
| Alur alternatif 4a. jika input yang dimasukkan tidak sesuai, maka <i>autosuggest</i> tidak akan muncul | |
| Alur alternatif 7a. jika input data yang dimasukkan tidak sesuai maka akan memunculkan pesan "Kata Kunci Pencarian Tidak Ditemukan". | |

Nama Use Case : Cari Nama

Aktor : User

Deskripsi : Proses ini adalah kegiatan untuk mencari keterangan tentang *perintah command line* tertentu dalam basisdata berdasarkan input kata kerja dasar tertentu.

Pre-Condition : User harus sudah berada di menu Pencarian

Post-Condition : Keterangan tentang perintah *command line* tertentu ditampilkan.

Tabel 12. Skenario Use Case Cari Nama

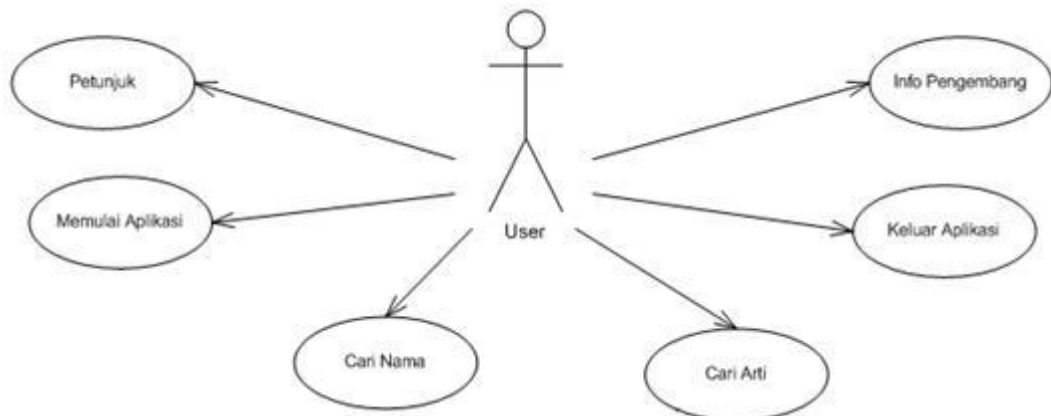
| Aksi Aktor | Reaksi Sistem |
|--|---|
| Alur Dasar | |
| 1. Menekan <i>menu tab</i> Cari Nama | |
| | 2. Menampilkan <i>menu tab</i> Cari Nama |
| 3. Memasukkan input kata kerja dasar tertentu tertentu | |
| 4. Menekan salah satu pilihan yang ditampilkan <i>autosuggest</i> . | |
| 5. Menekan tombol Cari Nama | |
| | 6. Mengecek dalam basisdata kata kerja dasar tertentu berdasarkan input. |
| | 7. Jika data yang dimasukkan ditemukan dalam basis data, maka data akan ditampilkan |
| 8. Menekan pilihan data yang ditampilkan | |
| | 9. Menampilkan <i>autosuggest</i> kata kerja dasar tertentu. |
| 10. Menekan salah satu pilihan yang ditampilkan <i>autosuggest</i> . | |

Tabel 13. Skenario Use Case Cari Nama(lanjutan)

| Aksi Aktor | Reaksi Sistem |
|--|--|
| 11. Menekan tombol Cari Nama | |
| | 12. Mengecek dalam basisdata kata kerja dasar tertentu berdasarkan input. |
| | 13. Jika data yang dimasukkan ditemukan dalam basis data, maka data akan ditampilkan |
| 14. Menekan pilihan data yang ditampilkan | |
| | 15. Menampilkan detail keterangan mengenai data perintah command line sesuai pilihan |
| Alur alternatif 4a. jika input yang dimasukkan tidak sesuai, maka <i>autosuggest</i> tidak akan muncul | |
| Alur alternatif 7a. jika input data yang dimasukkan tidak sesuai maka akan memunculkan pesan "Kata Kunci Pencarian Tidak Ditemukan". | |

c) Rancangan *Use Case*

Berdasarkan hasil deskripsi *actor* dan *use case* serta hasil skenario *use case* maka rancangan *use case* aplikasi yang akan dikembangkan adalah sebagai berikut :

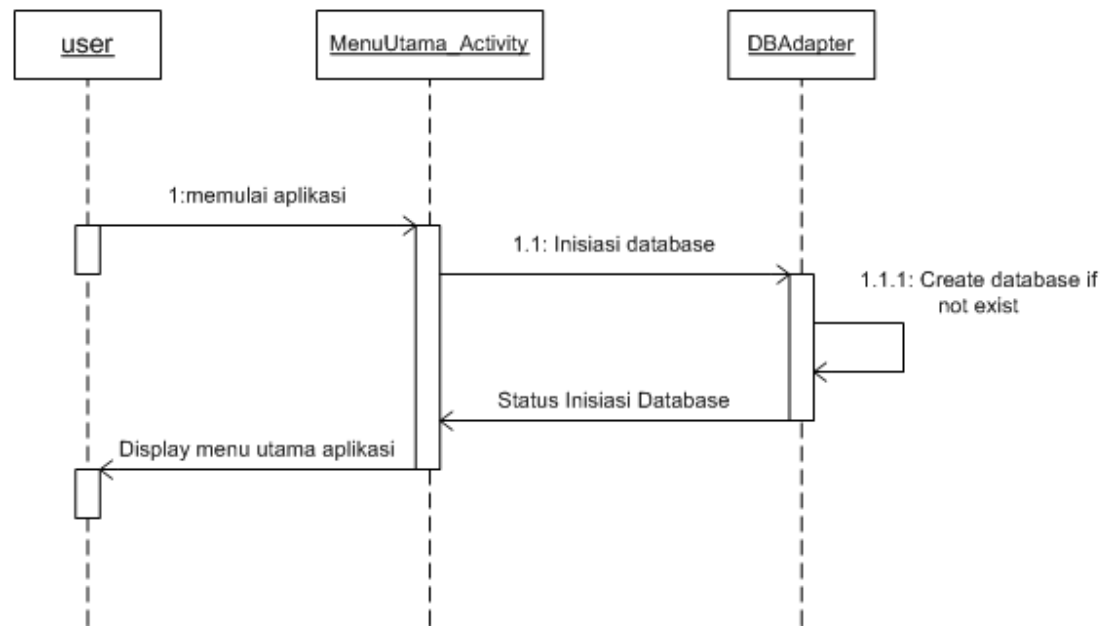


Gambar 8. Rancangan *Use Case* Diagram

2) *Sequence* Diagram

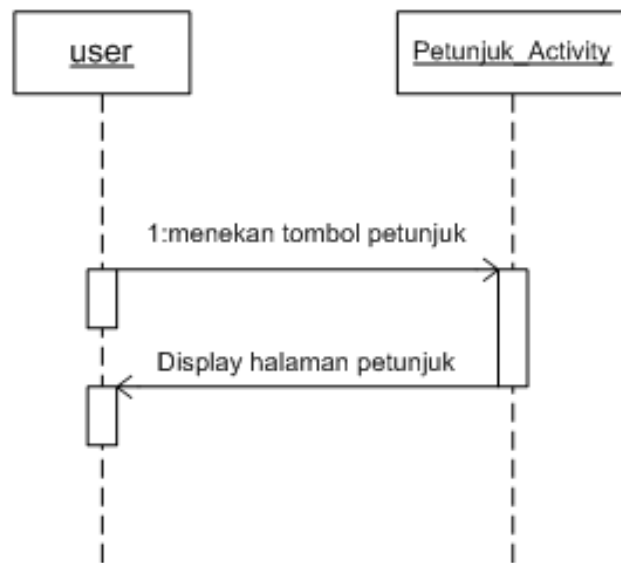
Sequence Diagram menggambarkan perilaku objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirim dan diterima antar objek. Jumlah *sequence* diagram yang dibuat sesuai dengan jumlah *use case* yang didefinisikan. *Sequence* diagram yang dibuat dalam tahap desain antara lain:

a) Diagram *Sequence* : Memulai Aplikasi



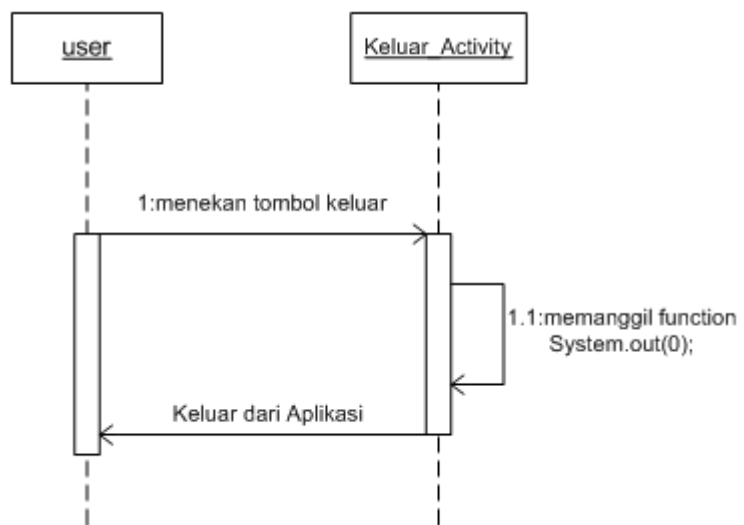
Gambar 9. *Sequence* Diagram Memulai Aplikasi

b) Diagram *Sequence* : Petunjuk



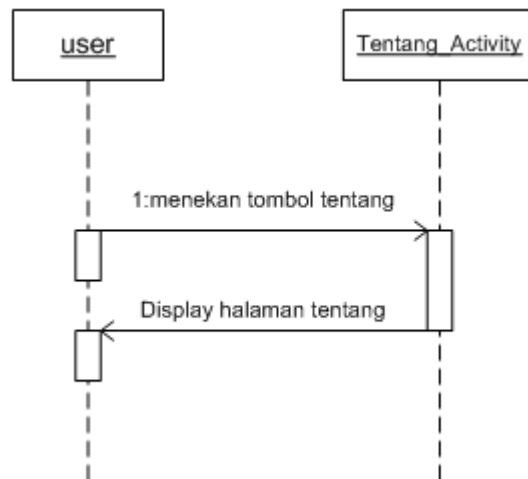
Gambar 10. *Sequence Diagram* Petunjuk

c) *Diagram Sequence* : Keluar Aplikasi



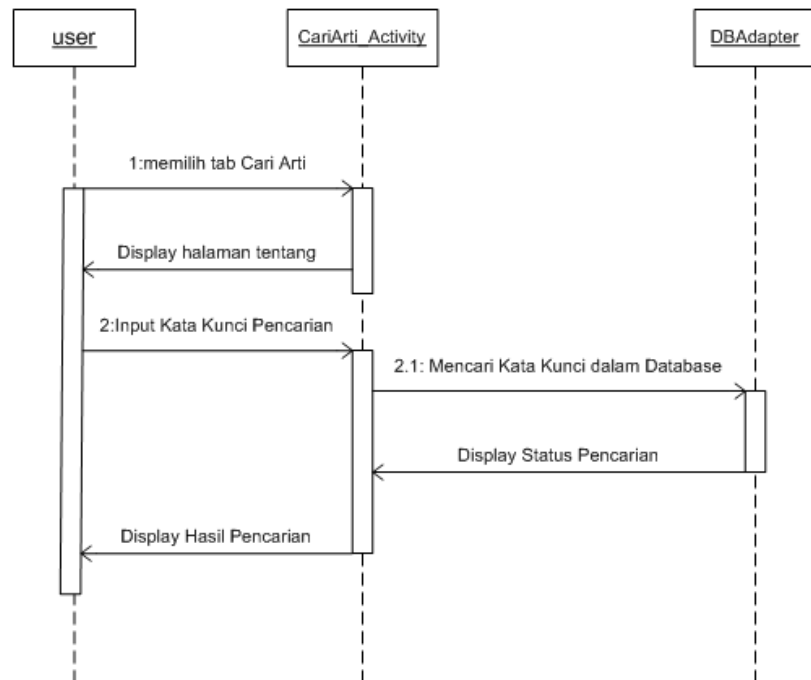
Gambar 11. *Sequence Diagram* Keluar Aplikasi

d) *Diagram Sequence* : Info Pengembang



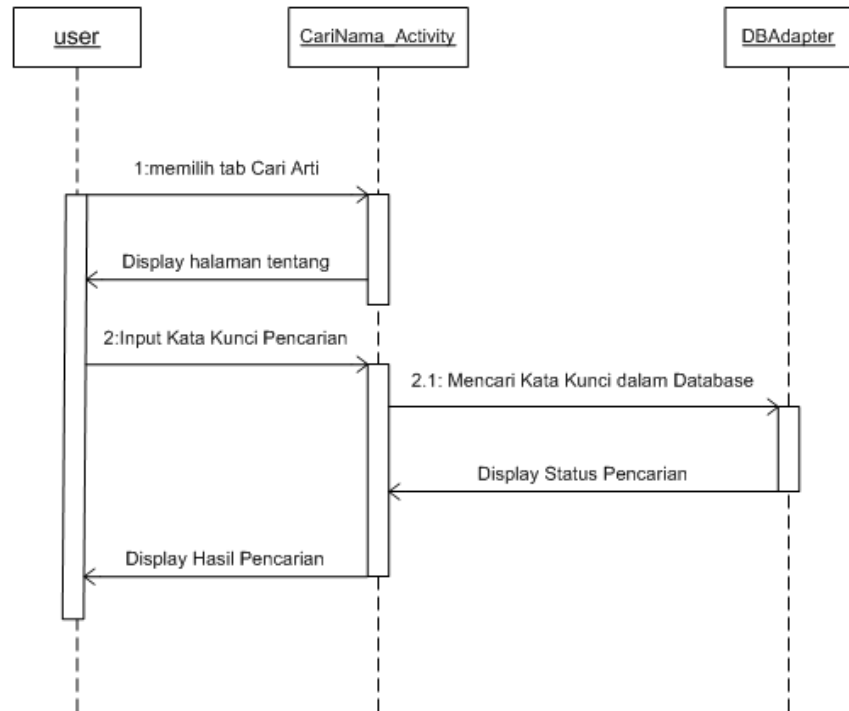
Gambar 12. *Sequence Diagram* Info Pengembang

e) *Diagram Sequence* : Cari Arti



Gambar 13. *Sequence Diagram* Cari Arti

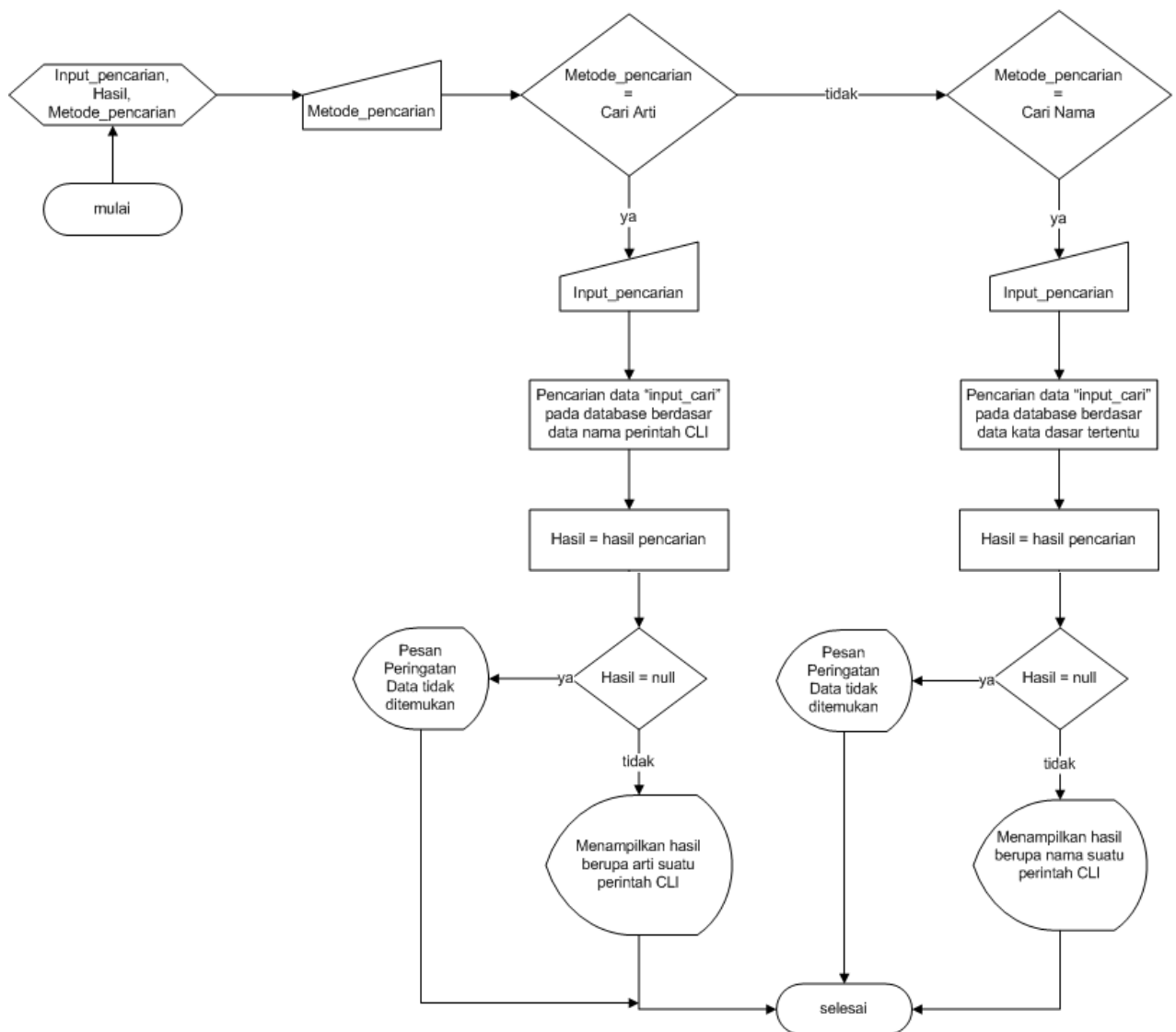
f) *Diagram Sequence* : Cari Nama



Gambar 14. *Sequence Diagram Cari Nama*

b. Perancangan Diagram Alir

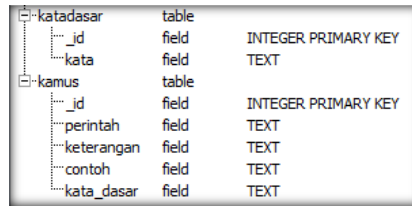
Diagram alir yang digunakan untuk menjelaskan aliran proses yang ada pada aplikasi. Diagram alir untuk proses pencarian kata ditunjukkan oleh gambar 15 dibawah ini



Gambar 15. Diagram Alir Proses Pencarian

c. Perancangan Basis data

Perancangan basis data pada sisi pengguna menggunakan *database SQLite* dengan *software SQLiteBrowser*. Terdapat dua tabel yaitu tabel katadasar dan tabel kamus. Struktur masing-masing tabel dijelaskan pada gambar 16 dibawah ini.

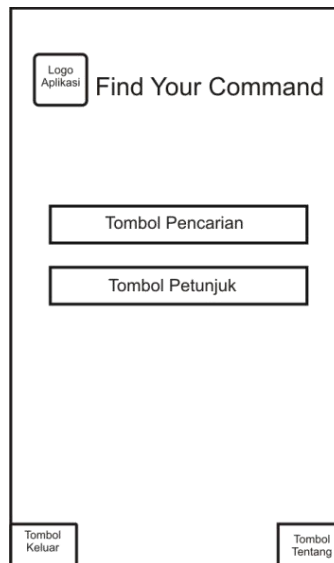


Gambar 16. Rancangan *Database* Tabel Kata dasar dan Kamus

d. Perancangan Antarmuka

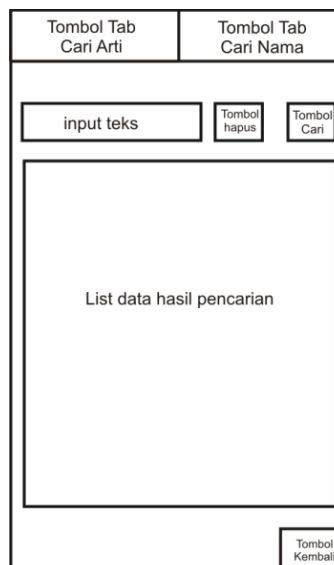
1) Perancangan Antarmuka

a) Rancangan antarmuka Halaman Menu Utama



Gambar 17. Rancangan Halaman Menu Utama

b) Rancangan Antarmuka Menu Pencarian



Gambar 18. Rancangan Halaman Menu Pencarian

c) Rancangan Antarmuka Detail Informasi

Wireframe of a detail information screen. It contains three input fields: "Nama Perintah" (short), "Keterangan Perintah" (long), and "Contoh Penggunaan Perintah" (short). A "Tombol Kembali" button is in the bottom right corner.

Gambar 19. Rancangan Halaman Detail Informasi

3. Implementasi

a. Persiapan Lingkungan pengembang

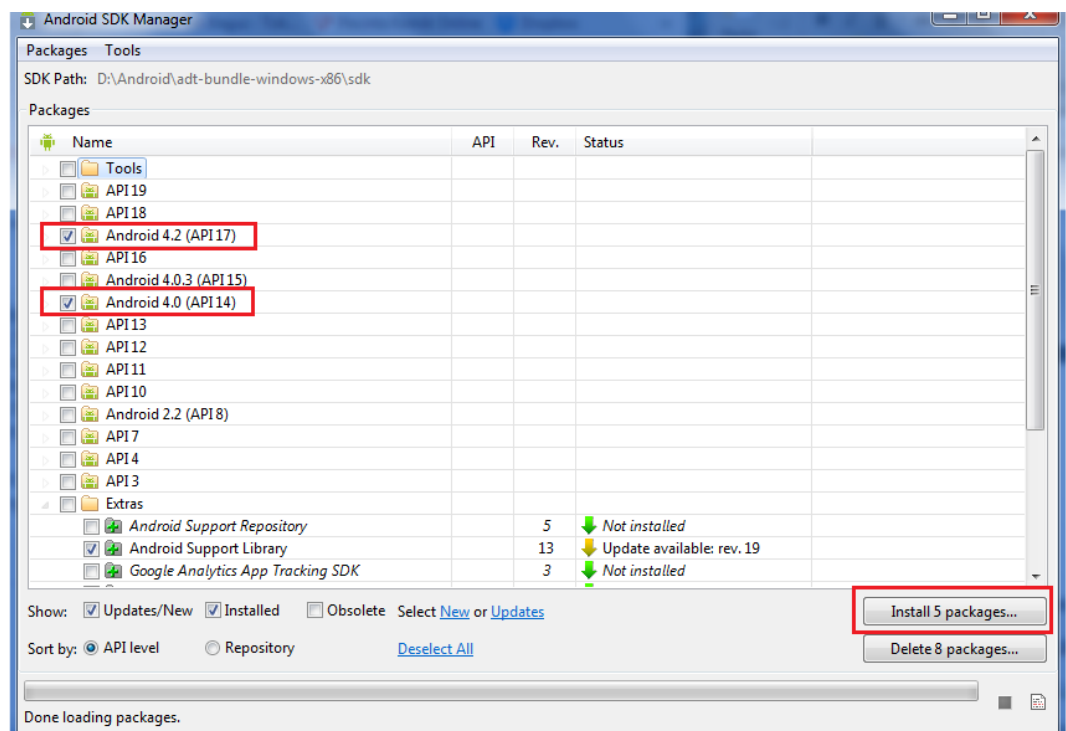
Proses implementasi dalam pengembangan perangkat lunak yang dilakukan oleh peneliti menggunakan *software bundle* yang disediakan oleh alamat web *developer.android.com*. *Software bundle* tersebut bernama *ADT Bundle for Windows*. *Software bundle* tersebut berisi *IDE Eclipse*, *Software Development Kit*, dan *SDK Manager*. Beberapa hal yang dipersiapkan peneliti untuk memulai proses implementasi yaitu :

1) *Update Application Programming Interface (API)*

Menurut Pocatilu (2010), *application programming interface* memungkinkan pengembang untuk mengakses *framework* yang mencakup seluruh komponen yang dipakai semua aplikasi android. *Framework* tersebut terdiri dari *application management*, *windows management*, *user interface graphical* dan *event handling*.

API standard yang ada pada *software bundle* adalah API 8. API yang perlu ditambahkan adalah API 14 dan API 17. API 14 akan digunakan untuk melakukan pengujian aspek portabilitas dari aspek instalability pada versi *android Ice Cream Sandwich*. API 17 akan digunakan untuk melakukan pengujian aspek portabilitas dari aspek instalability pada versi *android Jelly Bean*.

Proses *update API* menggunakan aplikasi *SDK Manager* yang sudah disediakan dalam *software bundle*. Komputer harus terhubung internet untuk melakukan *update*. Tampilan menu *SDK Manager* ditunjukkan gambar 20 dibawah ini.



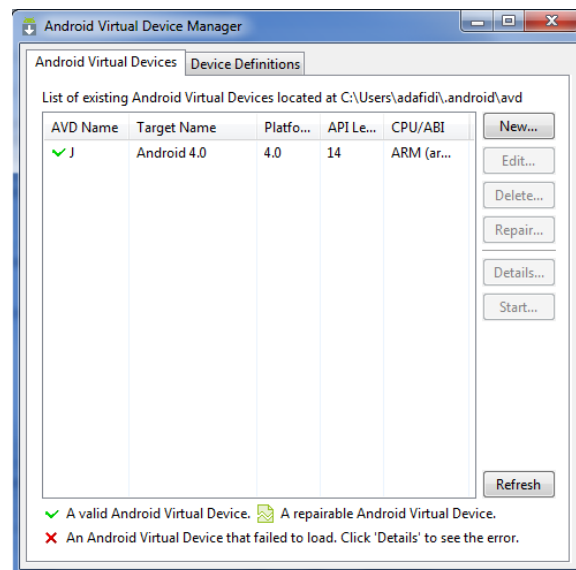
Gambar 20. Tampilan Menu *Android SDK Manager*

2) Membuat *Android Emulator*

Android emulator digunakan sebagai media untuk melakukan uji coba perangkat lunak yang dikembangkan. *Android emulator* yang

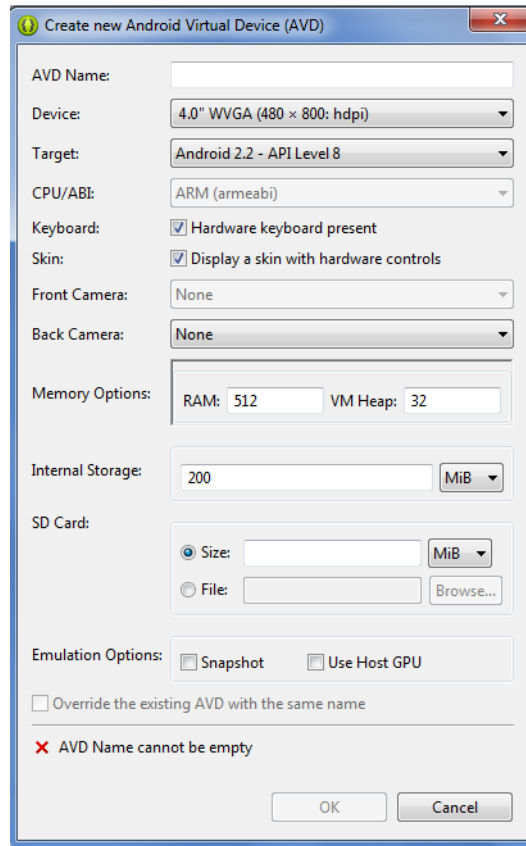
digunakan adalah *Android Virtual Device*. *Android Virtual Device* merupakan fitur yang sudah disediakan dalam *IDE Eclipse* yang berasal dari *software bundle*.

Proses membuat *android emulator* dimulai dengan membuka *menu bar windows* pada *IDE Eclipse* lalu memilih *Android Virtual Device Manager*. Tampilan menu fitur *Android Virtual Device Manager* ditunjukkan gambar 21 dibawah ini.



Gambar 21. Tampilan Menu *Android Virtual Device Manager*

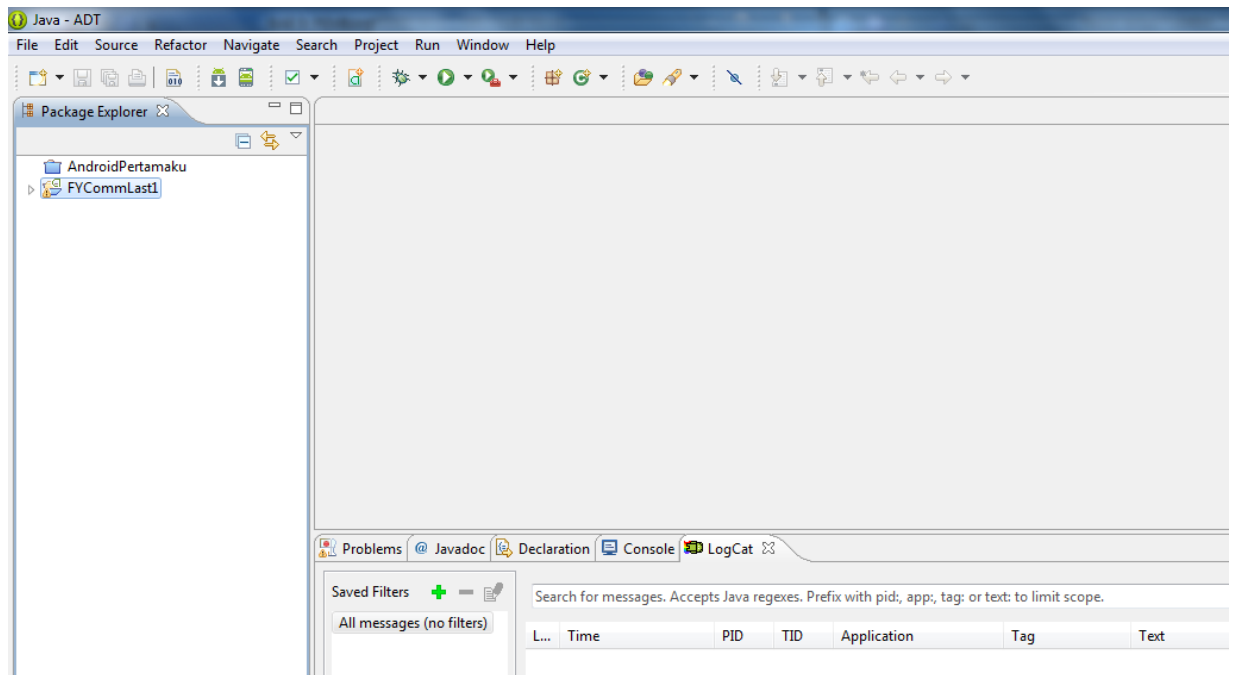
Langkah selanjutnya untuk membuat perangkat android virtual yaitu dengan menekan tombol *new*. Hal yang perlu diperhatikan dalam membuat perangkat *android virtual* yaitu pada bagian *device* dan target. *Device* diatur agar perangkat android virtual yang dibuat memiliki ukuran layar yang berbeda sedangkan *Target* diatur agar perangkat android menggunakan versi sistem *android* yang ditentukan. Untuk mengakhiri proses tekan tombol *Ok*. Tampilan menu *Create New AVD* ditunjukkan gambar 22 dibawah ini.



Gambar 22. Tampilan Menu *New* dalam *Android Virtual Device*

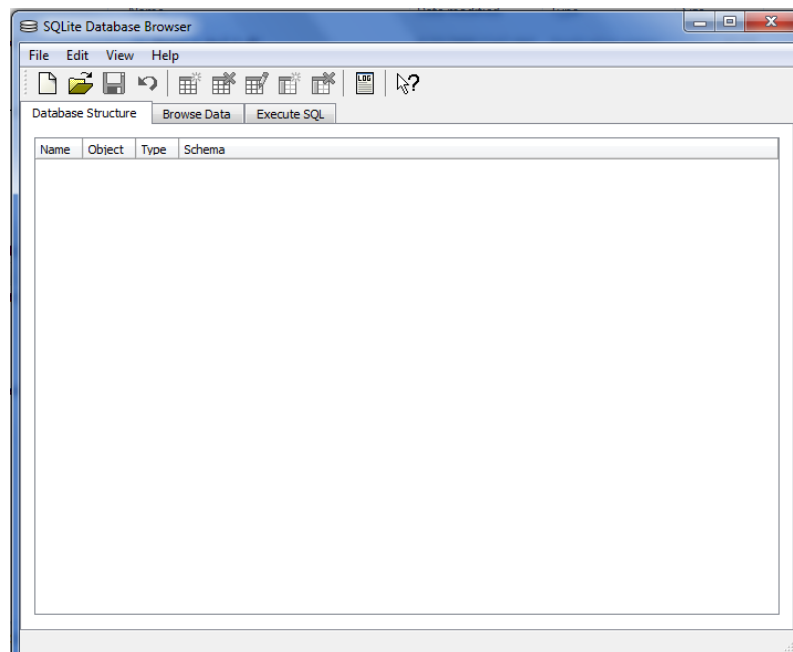
b. Implementasi pengkodean

Implementasi hasil desain ke dalam bentuk program akan menggunakan bahasa pemrograman *java*. Proses implementasi tersebut akan menggunakan *Integrated Development Environment (IDE) Eclipse*. *IDE Eclipse* yang digunakan berasal dari *software bundle* sehingga tidak perlu ada pengaturan lanjutan untuk memulai pemrograman *android*.



Gambar 23. Interface IDE Eclipse

Implementasi basis data akan menggunakan aplikasi *SQLitebrowser*. *SQLitebrowser* merupakan aplikasi yang dapat digunakan untuk membuat *SQLite databases*.



Gambar 24. Tampilan Aplikasi *SQLitebrowser*

4. Pengujian

Tahap yang dilakukan setelah tahap implementasi adalah tahap pengujian. Pengujian yang dilakukan pada tahap ini bertujuan untuk melakukan verifikasi dan validasi terhadap perangkat lunak yang dikembangkan. Verifikasi terkait dengan apakah perangkat lunak yang dikembangkan sesuai dengan spesifikasinya sedangkan validasi terkait dengan apakah perangkat lunak sudah sesuai dengan kriteria harapan pengguna (Sommerville, 2003:66). Proses verifikasi dilakukan dengan melakukan pengujian *white box*. Hal ini berdasarkan pendapat Roger.S. Pressman (1997:581), bahwa pengujian unit berfokus pada usaha verifikasi pada inti terkecil dari desain perangkat lunak dimana pengujian unit itu sendiri berorientasi pada pengujian *white box*.

Proses validasi pada perangkat yang dikembangkan akan dilakukan dengan melakukan pengujian *black box*, pengujian *alpha*, dan pengujian materi. Pemilihan pengujian *black box* dan pengujian *alpha* untuk menguji validitas program berdasarkan atas pendapat yang dikemukakan oleh Roger S.Pressman. Menurut Roger S.Pressman (2007:594), validasi perangkat lunak dicapai melalui sederetan pengujian *black box* yang memperlihatkan konformitas dengan persyaratan. Lebih lanjut lagi Roger S.Pressman (2007:595) menyatakan, bila perangkat lunak biasa dibangun bagi suatu pelanggan, maka sederetan *acceptance testing* dapat dilakukan untuk memungkinkan pelanggan memvalidasi perangkat lunak yang dikembangkan.

Acceptance testing dapat dilakukan melalui pengujian *alpha*. Pengujian *alpha* dapat dilakukan dengan menguji validitas fungsionalitas perangkat lunak kepada ahli dan menguji kualitas perangkat lunak yang

dikembangkan. Kualitas perangkat lunak yang dikembangkan diukur dalam beberapa *software quality factors*. *Software quality factors* yang digunakan mengacu pada standar yang dikeluarkan oleh ISO 9126. *Software quality factors* yang digunakan terdiri dari *functionality*, *efficiency*, *usability*, dan *portability*. Penentuan keempat kualitas perangkat lunak tersebut seperti yang diungkapkan pada bab II yaitu didasarkan pada analisis dua jurnal yang masing-masing ditulis oleh Aida Niknejad dan Assaf ben David.

Proses validasi lain yang dilakukan adalah pengujian materi. Uji materi dilakukan untuk memastikan materi yang disajikan dalam perangkat lunak teruji validitas kebenarannya. Pengujian materi dilakukan dengan memberikan kuesioner kepada ahli materi tentang sistem operasi Linux.

C. Variabel Penelitian

Variabel penelitian yang ada dalam penelitian ini adalah :

1. Functionality

Functionality merupakan aspek yang menunjukkan apakah produk mampu mengakomodasi apa yang diperlukan oleh pengguna. Variabel *functionality* diukur dari *sub* indikator *accuracy*.

2. Efficiency

Merupakan aspek yang terkait dengan kemampuan perangkat lunak untuk memberikan kinerja yang sesuai dan relatif terhadap jumlah sumber daya yang digunakan pada saat keadaan tersebut. Variabel *efficiency* diukur dari *sub* indikator *time behaviour*.

3. Usability

Usability merupakan aspek yang berkaitan dengan usaha yang diperlukan pengguna untuk mengoperasikan, menyiapkan *input*, dan

menginterpretasikan *output* program. Variabel *usability* diukur dari *sub* indikator *understandability*, *learnability*, dan *operatibility*.

4. Portability

Portability merupakan aspek yang berkaitan dengan kemampuan perangkat lunak untuk dapat dijalankan dalam lingkungan yang berbeda. Variabel *portability* diukur dari *sub* indikator *instalability* dan *adaptability*.

D. Tempat dan Waktu Penelitian

Tempat penelitian di Laboratorium Program Studi Pendidikan teknik Informatika Fakultas Teknik Universitas Yogyakarta untuk pengembangan, validasi dan revisi produk. Pengambilan data terkait variabel penelitian dilakukan di SMK N 1 Bantul. Waktu penelitian dimulai pada bulan Juni. Pemilihan waktu tersebut dilakukan karena mempertimbangkan agenda kegiatan sekolah seperti UAS, UAN, dan Ujian Kenaikan Kelas.

E. Subyek Penelitian

Salah satu tahap yang dilalui dalam penelitian *Pengembangan Aplikasi Android Kamus Command Line sebagai Media Bantu Belajar Siswa SMK Negeri 1 Bantul Kompetensi keahlian Teknik Komputer dan Jaringan* ini adalah tahap pengujian. Subyek penelitian yang digunakan pada penelitian ini terdiri dari ahli media, ahli materi, dan *Independent Testing Group*. Subyek penelitian ahli media digunakan peneliti untuk memvalidasi perangkat lunak yang dikembangkan berdasarkan penilaian oleh ahli media pendidikan. Subyek penelitian ahli materi digunakan peneliti untuk

memvalidasi materi yang ditampilkan perangkat lunak yang dikembangkan. Subyek penelitian *Independent Testing Group* berperan dalam pengujian *alpha* yang akan dilakukan oleh peneliti. Penggunaan *Independent Testing Group* bertujuan untuk mengetahui apakah perangkat lunak yang dikembangkan sudah sesuai dengan harapan pengguna akhir. Kesesuaian harapan tersebut diukur dari aspek kualitas perangkat lunak yang terdiri dari *functionality, efficiency, usability, dan portability*.

Penggunaan *Independent Testing Group* berdasarkan atas pendapat yang dikemukakan oleh Roger S.Pressman. Menurut Roger S.Pressman (1997:575), *Independent Testing Group (ITG)* berperan untuk menghilangkan masalah yang melekat sehubungan dengan diperbolehkannya pengembang untuk menguji perangkat lunak yang sudah dibangun. ITG beranggotakan 15 siswa SMK N 1 Bantul Kompetensi keahlian Teknik Komputer dan Jaringan. Keanggotaan ITG tersebut dipilih berdasarkan aspek pengguna akhir perangkat lunak nantinya.

F. Metode Pengumpulan Data

Metode yang digunakan untuk mengumpulkan data dalam penelitian ini yaitu :

1. Observasi

Menurut Sutrisno Hadi dalam Sugiyono (2007:203), observasi merupakan suatu proses yang kompleks, suatu proses yang tersusun dari pelbagai proses pengamatan dan ingatan. Teknik pengumpulan data dengan observasi dapat dilakukan bila penelitian berkenaan dengan perilaku manusia, proses kerja, gejala-gejala alam, dan bila responden yang diamati tidak terlalu besar. Metode Observasi dilakukan untuk melakukan pengujian

black box pada perangkat lunak. Metode observasi juga dilakukan untuk mendapatkan data yang terkait dengan aspek *portability*

2. Kuesioner

Menurut Sugiyono (2007:199), angket atau kuesioner merupakan teknik pengumpulan data yang dilakukan dengan cara memberi seperangkat pertanyaan atau pernyataan tertulis kepada responden untuk dijawab. Kuesioner dilakukan dengan menyebarkan beberapa pernyataan yang berhubungan dengan variabel penelitian kepada anggota sampel penelitian. Teknik kuesioner atau angket dilakukan untuk mendapatkan data yang terkait dengan aspek *functionality*, *efficiency*, dan *usability*. Teknik kuesioner juga dilakukan untuk melakukan uji materi dan uji validasi oleh ahli media pendidikan.

G. Instrumen Penelitian

Salah satu tahap penelitian yang dilakukan oleh peneliti adalah pengujian. Pengujian dilakukan untuk melakukan verifikasi dan validasi pada perangkat lunak yang dikembangkan. Tahap pengujian dalam rangka verifikasi dan validasi pada perangkat lunak menggunakan instrumen untuk memperoleh data. Instrumen yang digunakan yaitu observasi dan kuesioner. Jenis instrumen tersebut disesuaikan dengan teknik pengumpulan data yang digunakan.

Pengembangan instrumen yang dilakukan peneliti mengacu pada pendapat yang dikemukakan oleh Jogiyanto. Menurut Jogiyanto (2008:137), untuk membangun kuesioner dapat dilakukan melalui tiga tahap yaitu:

1. Melakukan pembentukan item

2. Melakukan *pretest* kepada ahli.
3. Menguji reliabilitas instrumen.

Instrumen penelitian yang telah dikembangkan peneliti sesuai dengan tahap diatas yaitu :

1. Lembar Observasi

Instrumen penelitian berupa lembar observasi digunakan dalam pengumpulan data yang dilakukan dengan teknik observasi. Lembar observasi digunakan dalam pengujian *black box* dan pengujian *alpha* untuk membantu mengumpulkan data yang terkait dengan pengujian aspek *portability*.

a. Pengujian Black Box

Pengujian *Black box* perangkat dilakukan dalam rangka melakukan tahapan validasi. Pengujian *Black box* perangkat dilakukan dengan tabel fungsionalitas yang menggunakan skala Guttman. Tabel fungsionalitas tersebut disusun berdasarkan *use case* yang dirancang pada tahap desain.

Pengujian *black box* dilakukan dengan bantuan *Android Virtual Device*. Mekanisme pengujian *black box* yang dilakukan yaitu dengan menguji daftar fungsionalitas yang dimiliki oleh perangkat lunak berdasarkan skenario *use case*.

b. Pengujian Kualitas Perangkat Lunak dari segi Portability

Pengambilan data untuk uji aspek *portability* dilakukan dengan bantuan aplikasi *android emulator*. Mekanisme pengujian yang akan dilakukan yaitu dengan menjalankan perangkat lunak pada *emulator* dengan versi sistem *android*, model tampilan, dan resolusi layar yang berbeda.

Lembar observasi yang digunakan untuk melakukan pengujian aspek *portability* ditunjukkan oleh tabel 14 dibawah ini :

Tabel 14. Lembar Observasi Pengujian Aspek *Portability*

| Variabel | Indikator | Sub Indikator | Hasil Yang diharapkan | Ketercapaian | |
|-------------|----------------|--|---|--------------|-------|
| | | | | Ya | Tidak |
| Portability | Installability | Keberhasilan dalam instalasi | Aplikasi <i>FyComm</i> berhasil diinstal pada android versi <i>Gingerbread</i> | | |
| | | | Aplikasi <i>FyComm</i> berhasil diinstal pada android versi <i>Ice Cream Sandwich</i> | | |
| | | | Aplikasi <i>FyComm</i> berhasil diinstal pada android versi <i>Jelly Bean</i> | | |
| | Adaptability | Penyesuaian terhadap jenis kerapatan layar | Aplikasi <i>FyComm</i> dapat dijalankan pada resolusi layar 480x800 dengan 240 dpi | | |
| | | | Aplikasi <i>FyComm</i> dapat dijalankan pada resolusi layar 320x480 dengan 160 dpi | | |
| | | | Aplikasi <i>FyComm</i> dapat dijalankan pada resolusi layar 240x320 dengan 120 dpi | | |
| | | Penyesuaian terhadap orientasi layar | Aplikasi dapat dijalankan pada orientasi <i>portrait</i> | | |
| | | | Aplikasi dapat dijalankan pada orientasi <i>landscape</i> | | |

2. Kuesioner

Instrumen penelitian berupa kuesioner digunakan dalam pengumpulan data yang dilakukan dengan teknik kuesioner. Kuesioner digunakan untuk mengumpulkan data terkait dengan pengujian *alpha* yang terdiri dari uji validasi oleh ahli media pendidikan dan uji kualitas perangkat lunak dari aspek *functionality*, *efficiency*, dan *usability*. Pengujian lain yang menggunakan instrumen kuesioner adalah pengujian materi.

a. Pengujian Alpha

1) Pengujian Validasi Oleh Ahli Media Pendidikan

Pengujian validasi oleh ahli media pendidikan dilakukan untuk memvalidasi perangkat lunak yang dikembangkan berdasarkan pendapat ahli media pendidikan. Ahli media pendidikan melakukan pengujian perangkat lunak untuk mengetahui permasalahan perangkat lunak pada lingkungan perspektif pengembang. Pada tahapan ini ahli media akan memberikan laporan berupa kesalahan-kesalahan yang terjadi serta usulan bagi pengembangan perangkat lunak. Pengujian dilakukan dengan mencocokkan daftar tabel spesifikasi perangkat lunak dengan unjuk kerja perangkat lunak yang dimiliki. Tabel-tabel yang digunakan yaitu :

Tabel 15. Spesifikasi Uji Membuka Aplikasi

| Aktivitas/Menu | Hasil Kerja yang Diharapkan | Ketercapaian | |
|------------------|--|--------------|-------|
| | | Ya | Tidak |
| Membuka Aplikasi | Tampil Halaman Menu Utama yang menampilkan 4(empat) tombol yaitu tombol pencarian, tombol petunjuk, tombol info pengembang, dan tombol keluar. | | |

Tabel 16. Spesifikasi Uji Menu Utama

| Aktivitas/Menu | Hasil Kerja yang Diharapkan | Ketercapaian | |
|----------------|---|--------------|-------|
| | | Ya | Tidak |
| Menu Utama | Menu Utama dapat terbuka ketika aplikasi pertama dijalankan. | | |
| | Halaman Pencarian dapat terbuka ketika <i>user</i> menekan tombol “Pencarian”. | | |
| | Halaman Petunjuk dapat terbuka ketika <i>user</i> menekan tombol “Petunjuk”. | | |
| | Halaman Identitas Pengembang dapat terbuka ketika <i>user</i> menekan tombol “Info Pengembang”. | | |
| | <i>User</i> dapat keluar aplikasi ketika menekan tombol “Keluar”. | | |

Tabel 17. Spesifikasi Uji Menu Pencarian

| Aktivitas/Menu | Hasil Kerja yang Diharapkan | Ketercapaian | |
|----------------|--|--------------|-------|
| | | Ya | Tidak |
| Menu Pencarian | Halaman Pencarian berupa tampilan <i>menu tab</i> Cari Arti dan <i>menu tab</i> Cari Nama dapat terbuka ketika <i>user</i> menekan tombol “Pencarian”. | | |
| | <i>User</i> dapat memilih dan masuk ke <i>menu tab</i> Cari Arti. | | |
| | <i>User</i> dapat memilih dan masuk ke <i>menu tab</i> Cari Nama. | | |

Tabel 18. Spesifikasi Uji Menu *Tab* Cari Arti

| Aktivitas/Menu | Hasil Kerja yang Diharapkan | Ketercapaian | |
|---------------------------|--|--------------|-------|
| | | Ya | Tidak |
| Menu <i>Tab</i> Cari Arti | <i>User</i> dapat kembali ke menu utama dengan menekan tombol “Home” pada halaman <i>menu tab</i> Cari Arti. | | |
| | <i>Input text</i> pada <i>menu tab</i> Cari Arti dapat berfungsi dengan baik. | | |

Tabel 19. Spesifikasi Uji Menu *Tab* Cari Arti(lanjutan)

| Aktivitas/Menu | Hasil Kerja yang Diharapkan | Ketercapaian | |
|---------------------------|--|--------------|-------|
| | | Ya | Tidak |
| Menu <i>Tab</i> Cari Arti | <i>User</i> dapat kembali ke menu utama dengan menekan tombol “Home” pada halaman <i>menu tab</i> Cari Arti. | | |
| | <i>Input text</i> pada <i>menu tab</i> Cari Arti dapat berfungsi dengan baik. | | |
| | Fungsi <i>auto suggestion</i> berupa perintah <i>command line</i> pada <i>input text</i> berfungsi dengan baik. | | |
| | <i>User</i> dapat menghilangkan tulisan pada <i>input text</i> dengan menekan tombol “Hapus”. | | |
| | <i>User</i> dapat menampilkan hasil pencarian dengan menekan tombol “Cari”. | | |
| | Hasil pencarian aplikasi sesuai dengan kata kunci pencarian. | | |
| | Aplikasi akan memunculkan pesan ketika hasil pencarian tidak ditemukan. | | |
| | <i>User</i> dapat menampilkan detail informasi perintah <i>command line</i> dengan menekan salah satu hasil pencarian. | | |

Tabel 20. Spesifikasi Uji Menu *Tab* Cari Nama

| Aktivitas/Menu | Hasil Kerja yang Diharapkan | Ketercapaian | |
|---------------------------|--|--------------|-------|
| | | Ya | Tidak |
| Menu <i>tab</i> Cari Nama | <i>User</i> dapat kembali ke menu utama dengan menekan tombol “Home” pada halaman <i>menu tab</i> Cari Nama. | | |
| | <i>Input text</i> pada <i>menu tab</i> Cari Nama dapat berfungsi dengan baik. | | |

Tabel 21 . Spesifikasi Uji Menu *Tab* Cari Nama(lanjutan)

| Aktivitas/Menu | Hasil Kerja yang Diharapkan | Ketercapaian | |
|---------------------------|--|--------------|-------|
| | | Ya | Tidak |
| Menu <i>tab</i> Cari Nama | <i>User</i> dapat kembali ke menu utama dengan menekan tombol “ <i>Home</i> ” pada halaman <i>menu tab</i> Cari Nama. | | |
| | <i>Input text</i> pada <i>menu tab</i> Cari Nama dapat berfungsi dengan baik. | | |
| | Fungsi <i>auto suggestion</i> berupa kata kerja dasar tertentu pada <i>input text</i> berfungsi dengan baik. | | |
| | <i>User</i> dapat menghilangkan tulisan pada <i>input text</i> dengan menekan tombol “Hapus”. | | |
| | <i>User</i> dapat menampilkan hasil pencarian dengan menekan tombol “Cari”. | | |
| | Hasil pencarian aplikasi sesuai dengan kata kunci pencarian. | | |
| | Aplikasi akan memunculkan pesan ketika hasil pencarian tidak ditemukan. | | |
| | <i>User</i> dapat menampilkan detail informasi perintah <i>command line</i> dengan menekan salah satu hasil pencarian. | | |

Tabel 22. Spesifikasi Uji Menu Petunjuk

| Aktivitas/Menu | Hasil Kerja yang Diharapkan | Ketercapaian | |
|----------------|--|--------------|-------|
| | | Ya | Tidak |
| Menu Petunjuk | <i>User</i> dapat melihat halaman petunjuk penggunaan aplikasi ketika menekan tombol “Petunjuk” pada menu utama. | | |
| | <i>User</i> dapat melihat petunjuk penggunaan aplikasi sebelumnya dengan menggeser layar ke kanan. | | |
| | <i>User</i> dapat melihat petunjuk penggunaan aplikasi selanjutnya dengan menggeser layar ke kiri. | | |
| | <i>User</i> dapat kembali ke menu utama dengan menekan tombol “ <i>Home</i> ” pada layar. | | |

Tabel 23. Spesifikasi Uji Menu Info Pengembang

| Aktivitas/Menu | Hasil Kerja yang Diharapkan | Ketercapaian | |
|----------------------|--|--------------|-------|
| | | Ya | Tidak |
| Menu Info Pengembang | User dapat melihat informasi tentang identitas pengembang ketika menekan tombol “Info Pengembang”. | | |
| | User dapat kembali ke menu utama dengan menekan tombol “Home” pada layar. | | |

2) Pengujian Kualitas Perangkat Lunak dari segi *functionality*, *efficiency*, dan *usability*.

Pengujian kualitas perangkat lunak dilakukan dalam rangka melakukan validasi pada perangkat lunak. Kuesioner yang digunakan untuk pengujian kualitas perangkat lunak berupa butir-butir instrumen. Butir-butir instrumen tersebut berdasarkan kriteria dari *software quality factors* ISO 9126.

Kualitas perangkat lunak tersebut diukur dari aspek *functionality*, *efficiency*, dan *usability*. Instrumen kuesioner ini nantinya akan divalidasi oleh validator dengan metode *expert judgment*. Validasi instrumen tersebut dilakukan agar data yang dihasilkan nantinya dapat teruji validitasnya.

Kuesioner yang akan digunakan pada pengujian kualitas perangkat lunak menggunakan kisi-kisi sebagai berikut :

Tabel 24. Kisi-Kisi Instrumen Pengujian *Functionality*, *Efficiency*, dan *Usability*

| Variabel | Indikator | Sub Indikator | Item Butir |
|----------------------|--------------------------|--|---------------|
| <i>Functionality</i> | <i>Suitability</i> | Semua fitur berjalan dengan baik. | 1,2,3,4,5,6,7 |
| <i>Efficiency</i> | <i>Time Behaviour</i> | Lama waktu tiap operasi.. | 8,9,10 |
| <i>Usability</i> | <i>Understandability</i> | Kejelasan informasi yang diberikan aplikasi. | 11 |
| | | Pemberian informasi yang mudah dipelajari. | 12,13 |
| | <i>Learnability</i> | Kemudahan dalam mempelajari <i>interface</i> . | 14,15 |
| | | Kemudahan dalam mengakses petunjuk penggunaan. | 16 |
| | <i>Operatibility</i> | Kenyamanan dalam penggunaan. | 17 |
| | | Kepuasan dalam penggunaan. | 18 |
| <i>Portability</i> | <i>Instalability</i> | Keberhasilan dalam instalasi. | 17,18,19 |
| | <i>Adaptability</i> | Penyesuaian terhadap jenis kerapatan layar device. | 20,21,22 |
| | | Penyesuaian terhadap orientasi layar. | 23,24 |

Kisi-kisi instrumen pengujian kualitas perangkat lunak seperti yang digambarkan pada tabel 24 akan digunakan sebagai dasar pembuatan instrumen pengujian kualitas perangkat lunak. Kisi-kisi tersebut digunakan untuk membuat pernyataan-pernyataan yang mewakili setiap variabel yang akan diuji dalam pengujian kualitas perangkat lunak. Kuesioner yang akan digunakan dalam pengujian kualitas perangkat lunak dijabarkan pada tabel 25 dan 26 dibawah ini :

Tabel 25. Instrumen Pengujian *Functionality*, *Efficiency*, dan *Usability*

| No | Pernyataan | Alternatif Jawaban | | | | |
|-------------------------------|--|--------------------|----|---|---|----|
| | | STS | TS | N | S | SS |
| Variabel <i>Functionality</i> | | | | | | |
| 1. | Perangkat lunak dapat melakukan pencarian berdasarkan metode pencarian Cari Nama. | | | | | |
| 2. | Perangkat lunak dapat melakukan pencarian berdasarkan metode pencarian Cari Arti. | | | | | |
| 3. | Perangkat lunak dapat menampilkan detail informasi perintah dari hasil metode pencarian Cari Nama. | | | | | |
| 4. | Perangkat lunak dapat menampilkan detail informasi perintah dari metode pencarian Cari Arti. | | | | | |
| 5. | Perangkat lunak dapat menampilkan halaman informasi pengembang aplikasi. | | | | | |
| 6. | Metode pencarian Cari Nama menghasilkan hasil yang sesuai dengan kata kunci yang dimasukkan. | | | | | |
| 7. | Metode pencarian Cari Arti menghasilkan hasil yang sesuai dengan kata kunci yang dimasukkan. | | | | | |
| Variabel <i>Efficiency</i> | | | | | | |
| 8. | Proses pencarian pada metode pencarian Cari Nama membutuhkan waktu yang singkat. | | | | | |
| 9. | Proses pencarian pada metode pencarian Cari Arti membutuhkan waktu yang singkat. | | | | | |
| 10. | Saya puas dengan durasi respon pada setiap fitur aplikasi. | | | | | |
| Variabel <i>Usability</i> | | | | | | |
| 11. | Aplikasi ini memberikan informasi dengan jelas. | | | | | |
| 12. | Informasi yang disediakan dapat membantu menyelesaikan tugas saya. | | | | | |

Tabel 26. Instrumen Pengujian *Functionality*, *Efficiency*, dan *Usability*(lanjutan)

| No | Pernyataan | Alternatif Jawaban | | | | |
|---------------------------|---|--------------------|----|---|---|----|
| | | STS | TS | N | S | SS |
| Variabel <i>Usability</i> | | | | | | |
| 13. | Informasi yang disediakan aplikasi ini memudahkan saya dalam belajar. | | | | | |
| 14. | <i>Interface</i> aplikasi ini mudah dipelajari. | | | | | |
| 15. | Saya dapat mempelajari penggunaan aplikasi ini dengan mudah. | | | | | |
| 16. | Aplikasi ini menyediakan menu petunjuk penggunaan. | | | | | |
| 17. | Secara keseluruhan aplikasi ini nyaman ketika digunakan. | | | | | |
| 18. | Kemudahan dalam menggunakan aplikasi ini membuat saya puas. | | | | | |

b. Uji Materi

Uji materi dilakukan dalam rangka melakukan validasi pada perangkat lunak. Instrumen penelitian berupa kuesioner pada pengujian *alpha* dilakukan untuk pengujian materi. Pengujian materi dilakukan untuk memvalidasi materi yang disajikan perangkat lunak. Materi tersebut berupa nama perintah *command line*, arti perintah *command line*, dan contoh penggunaan perintah *command line* tertentu.

Kuesioner yang akan diujikan berupa daftar nama perintah *command line*, arti perintah *command line*, dan contoh penggunaan *command line*. Responden untuk pengujian validasi materi adalah dua orang ahli materi dibidang sistem operasi Linux. Kuesioner yang akan digunakan untuk melakukan uji materi terlampir

H. Teknik Analisis Data

Teknik analisis yang digunakan dalam penelitian ini menggunakan teknik analisis skala Likert. Proses analisis ini digunakan untuk menghitung data variabel terdiri dari *functionality*, *efficiency*, *usability*, dan *portability*. Perhitungan yang digunakan untuk mengolah data hasil instrumen yaitu perhitungan nilai rata-rata dan perhitungan persentasi skor tiap variabel. Rumus perhitungan rata-rata instrumen yang digunakan yaitu :

$$\bar{x} = \frac{\sum x}{n}$$

Keterangan :

\bar{x} = Skor rata-rata

$\sum x$ = Skor total item

n = Jumlah item

Rumus perhitungan untuk menghitung persentase skor yaitu :

$$\text{Persentase Kelayakan} = \frac{\text{Skor yang diobservasi}}{\text{Skor yang diharapkan}} \times 100\%$$

Persentase kelayakan yang didapat dari perhitungan tersebut kemudian dikonversi ke dalam pernyataan predikat. Proses konversi persentase kelayakan ke dalam pernyataan predikat menggunakan tabel Skala Likert. Konversi persentase ke pernyataan seperti dalam tabel seperti berikut (Riduwan & Sunarto, 2012:23):

Tabel 27. Interpretasi Persentase Likert

| No | Persentasi | Interpretasi |
|----|------------|--------------|
| 1. | 0% - 20% | Sangat Lemah |
| 2. | 21% - 40% | Lemah |
| 3. | 41% - 60% | Cukup |
| 4. | 61% - 80% | Kuat |
| 5. | 81% - 100% | Sangat kuat |

Nilai interpretasi yang pada tabel interpretasi persentase Likert tersebut akan disesuaikan dengan penelitian yang dilakukan. Penyesuaian interpretasi tersebut dikarenakan penelitian ini melakukan uji kelayakan perangkat lunak yang dikembangkan. Skala konversi persentasi yang sudah disesuaikan ditunjukan pada tabel 28 dibawah ini :

Tabel 28. Penyesuaian Interpretasi Likert

| No | Persentasi | Interpretasi |
|----|------------|--------------------|
| 1. | 0% - 20% | Sangat Tidak Layak |
| 2. | 21% - 40% | Tidak Layak |
| 3. | 41% - 60% | Cukup Layak |
| 4. | 61% - 80% | Layak |
| 5. | 81% - 100% | Sangat Layak |

Proses konversi data kuantitatif tersebut akan mendapatkan interpretasi kelayakan perangkat lunak yang dikembangkan berdasarkan aspek yang telah ditentukan. Hasil penelitian ini nantinya akan menentukan kualitas perangkat lunak baik per faktor kualitas maupun secara keseluruhan.

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

A. Hasil Penelitian

1. Implementasi

a. Implementasi Pemrograman

Tahap-tahap yang dilakukan pada proses implementasi disesuaikan berdasarkan hasil pada tahap desain. Hal tersebut bertujuan agar perangkat lunak yang dihasilkan nantinya sesuai dengan spesifikasi yang dibutuhkan oleh pengguna akhir. Implementasi pemrograman yang dilakukan oleh peneliti menggunakan *Integrated Development Environment Eclipse*. Bahasa pemrograman yang digunakan adalah bahasa pemrograman *java*.

Proses implelementasi pemrograman yang dilakukan peneliti menghasilkan beberapa kelas program. Kelas program tersebut memiliki fungsi-fungsi tersendiri. Kelas-kelas program yang dihasilkan pada tahap implementasi pemrograman antara lain kelas *MenuUtama*, kelas *HalamanTentang*, kelas *HalamanPetunjuk*, kelas *ImageAdapter*, kelas *Nama_Info*, kelas *Arti_Info*, Kelas *Cari_Arti*, kelas *Cari_Nama*, kelas *DBAdapter*, dan kelas *HalamanCari*.

1) Kelas MenuUtama

Kelas *MenuUtama* digunakan untuk mengatur semua hal yang ditampilkan pada halaman utama aplikasi. Kelas *MenuUtama* bertugas sebagai kelas yang mengatur halaman *layout* *menu_utama.xml*. Kelas *MenuUtama* juga bertugas untuk melakukan inisiasi

hubungan dengan *database* datakamus. Selain itu, kelas Menu-Utama juga mengatur fungsi tombol-tombol yang ada pada *layout* menu_utama.xml. Tombol-tombol tersebut adalah tombol cari, tombol petunjuk, tombol tentang, dan tombol keluar.

2) Kelas DBAdapter

Kelas DBAdapter digunakan untuk mengatur semua hal yang berkaitan dengan *database* aplikasi. Kelas DBAdapter mengatur hal-hal seperti proses inisiasi hubungan *database* dengan aplikasi, proses menampilkan perintah tertentu, dan proses menampilkan informasi detail perintah tertentu.

3) Kelas HalamanTentang

Kelas HalamanTentang digunakan untuk mengatur hal-hal yang akan ditampilkan berkaitan dengan tampilan identitas pengembang aplikasi. Kelas HalamanTentang bertugas untuk mengatur halaman *layout* halaman_tentang.xml. Kelas HalamanTentang juga mengatur fungsi tombol untuk kembali ke menu utama.

4) Kelas HalamanPetunjuk

Kelas HalamanPetunjuk bertugas untuk mengatur halaman *layout* halaman_petunjuk.xml. Kelas HalamanPetunjuk juga bertugas untuk mengatur fungsi tombol kembali ke menu utama. Selain itu Kelas HalamanPetunjuk digunakan untuk mengatur hal-hal yang akan ditampilkan berkaitan dengan petunjuk penggunaan aplikasi. Petunjuk penggunaan aplikasi ditampilkan dengan gambar yang diatur dalam kelas ImageAdapter.

5) Kelas ImageAdapter

Kelas Image Adapter digunakan untuk mengatur gambar-gambar yang akan ditampilkan sebagai petunjuk penggunaan aplikasi.

6) Kelas HalamanCari

Kelas HalamanCari digunakan untuk mengatur tampilan halaman pencarian aplikasi. Kelas HalamanCari bertugas untuk mengatur halaman *layout* halaman_cari.xml. Halaman pencarian menggunakan objek menu tab yang terdiri dari dua menu yaitu Cari Arti dan Cari Nama. Menu Cari Arti berhubungan dengan kelas Cari_Arti dan menu Cari Nama berhubungan dengan kelas Cari_Nama.

7) Kelas Cari_Arti

Kelas Cari_Arti digunakan untuk mengatur semua hal yang berkaitan dengan proses pencarian informasi perintah berdasarkan kata kunci nama perintah *command line* tertentu. Kelas Cari_Arti bertugas mengatur halaman *layout* cariarti.xml. Kelas Cari_Arti juga mengatur fungsi tombol-tombol yang ada pada layout cariarti.xml. Tombol-tombol tersebut adalah tombol cari, tombol hapus, dan tombol kembali.

8) Kelas Cari_Nama

Kelas Cari_Nama digunakan untuk mengatur semua hal yang berkaitan dengan proses pencarian informasi perintah berdasarkan kata kunci kata kerja tertentu. Kelas Cari_Nama bertugas mengatur halaman *layout* carinama.xml. Kelas Cari_Nama juga mengatur fungsi tombol-tombol yang ada pada layout carinama.xml. Tombol-

tombol tersebut adalah tombol cari, tombol hapus, dan tombol kembali.

9) Kelas Nama_Info

Kelas Nama_Info digunakan untuk mengatur semua hal yang berkaitan dengan proses menampilkan informasi detail mengenai perintah tertentu berdasarkan hasil pencarian pada metode Cari Arti. Informasi yang ditampilkan berupa nama perintah *command line*, keterangan perintah *command line*, dan contoh penggunaan perintah *command line* tertentu. Kelas Nama_Info juga bertugas untuk mengatur fungsi tombol kembali yang ada pada *layout nama_info.xml*.

10) Kelas Arti_Info

Kelas Arti_Info digunakan untuk mengatur semua hal yang berkaitan dengan proses menampilkan informasi detail mengenai perintah tertentu berdasarkan hasil pencarian pada metode Cari Nama. Informasi yang ditampilkan berupa nama perintah *command line*, keterangan perintah *command line*, dan contoh penggunaan perintah *command line* tertentu. Kelas Arti_Info juga bertugas untuk mengatur fungsi tombol kembali yang ada pada *layout arti_info.xml*.

b. Implementasi Antarmuka

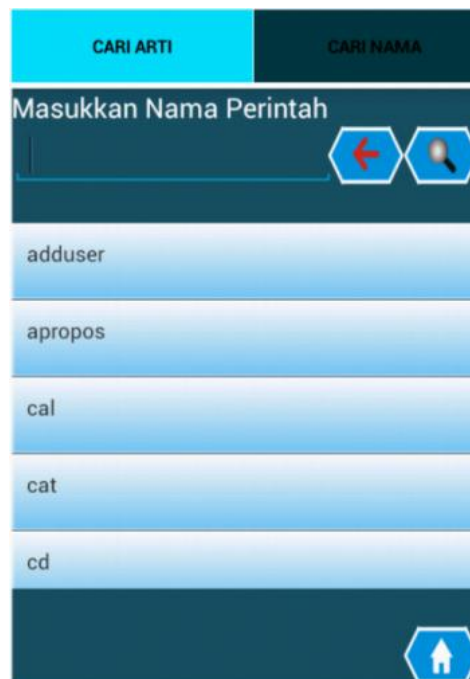
Implementasi antarmuka yang dilakukan berdasarkan hasil rancangan antar muka pada tahap desain. Hasil dari implementasi antarmuka yaitu :

1) Implementasi antarmuka Halaman Menu Utama



Gambar 25. Antarmuka Halaman Utama

2) Implementasi antarmuka Halaman Menu Pencarian



Gambar 26. Antarmuka Halaman Pencarian

3) Implementasi antarmuka Halaman Detail Informasi



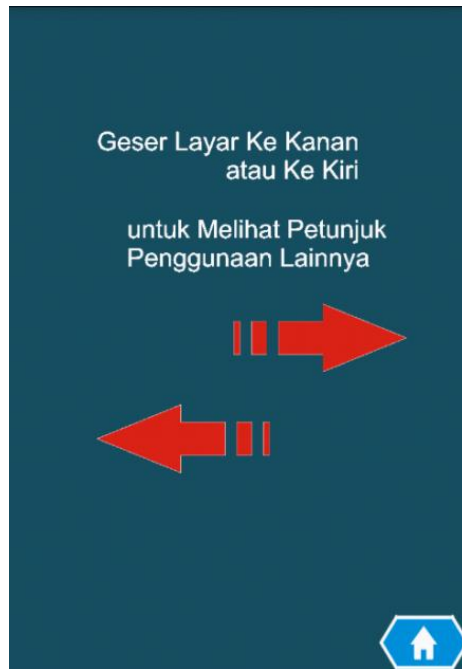
Gambar 27. Antarmuka Halaman Detail Informasi

4) Implementasi antarmuka Halaman Tentang



Gambar 28. Antarmuka Halaman Informasi Pengembang

5) Implementasi antarmuka Halaman Petunjuk



Gambar 29. Antarmuka Halaman Petunjuk

2. Pengujian Perangkat Lunak

a. Pengujian *White Box*

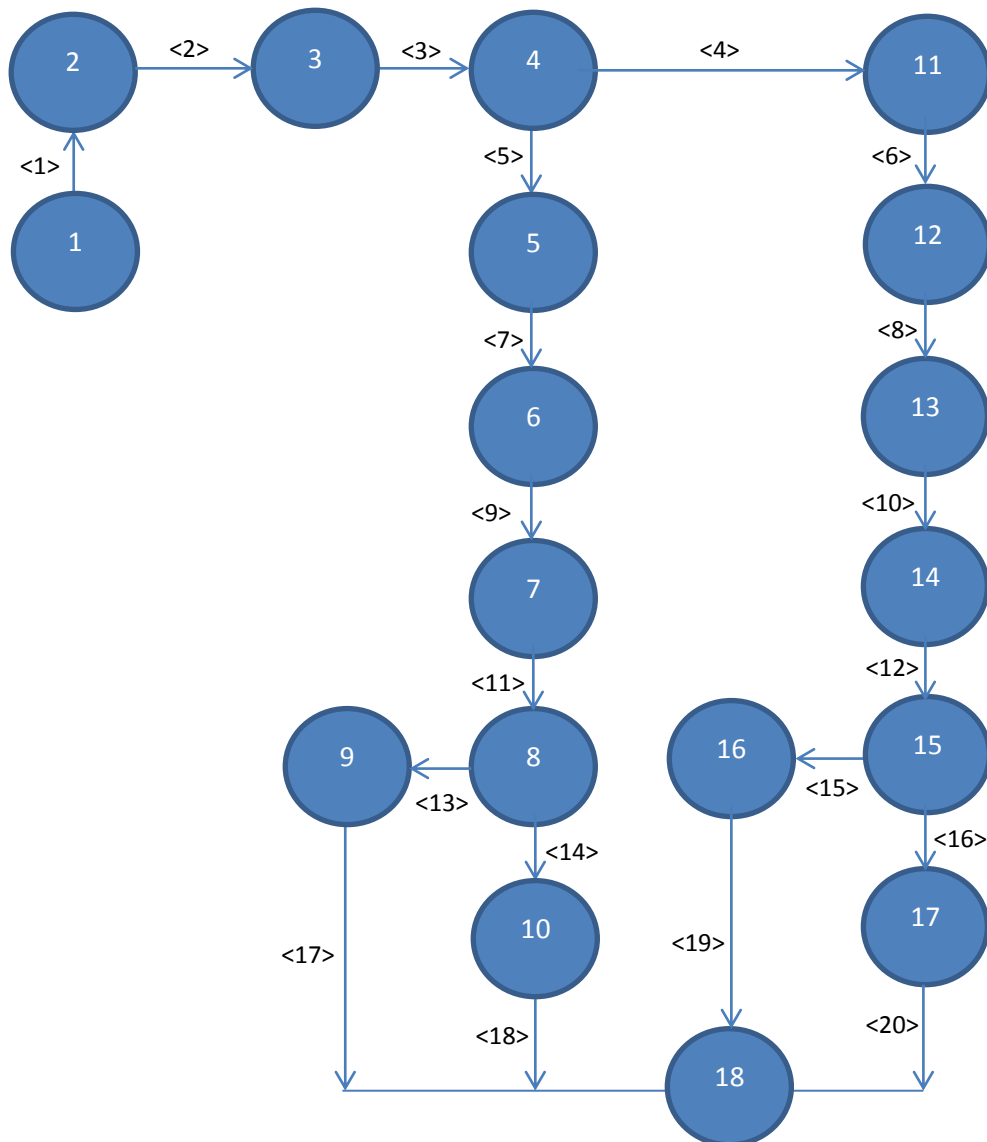
Pengujian *white box* dilakukan dengan metode *basis path testing*. Pengujian dengan metode *basis path testing* dilakukan dengan mengeksekusi seluruh jalur independen aplikasi minimal satu kali. Jalur independen program tersebut ditentukan melalui analisa pada notasi diagram alir. Jumlah jalur independen ditentukan melalui metode perhitungan *Cyclomatic Complexity*. Tahapan pengujian *white box* yang dilakukan peneliti yaitu :

1) Menentukan Notasi Diagram Alir

Pembuatan notasi diagram alir yang dilakukan oleh peneliti mengacu pada pendapat Roger S.Pressman (2002:536). Notasi diagram alir yang ditunjukkan oleh gambar 30 dibawah ini merupakan

notasi diagram alir aplikasi yang dibuat berdasarkan diagram alir (lihat gambar 15) aplikasi yang dikembangkan.

Pada gambar 30 dibawah ini *node* digambarkan sebagai lingkaran dengan angka didalamnya sedangkan *edge* digambarkan sebagai garis dengan anak panah yang menghubungkan antar *node*.



Gambar 30. Notasi Diagram Alir Proses Pencarian

Berdasarkan gambar diatas dapat diketahui jumlah *edge* yaitu 20 dan jumlah *node* yaitu 18. Seluruh *node* yang ditunjukkan pada gambar 30 tersebut mewakili seluruh simbol pada diagram alir (*flowchart*) aplikasi. Simbol pada diagram alir itu sendiri menggambarkan setiap proses yang ada pada aplikasi. Penjelasan terhadap *node-node* tersebut yaitu :

a) Node 1

Node 1 pada notasi diagram alir mewakili simbol *terminator* pada diagram alir aplikasi. *Node 1* mewakili tahapan permulaan (*start*) untuk proses pencarian pada aplikasi.

b) Node 2

Node 2 pada notasi diagram alir mewakili simbol *preparation* pada diagram alir aplikasi. *Node 2* mewakili tahapan persiapan pada proses pencarian aplikasi. Persiapan yang dilakukan berupa inisiasi variabel *input_pencarian*, *hasil_pencarian*, dan *metode_pencarian*.

c) Node 3

Node 3 pada notasi diagram alir mewakili simbol manual input pada diagram alir aplikasi. *Node 3* mewakili proses input manual untuk mendapatkan data dari *user* terkait dengan metode pencarian yang digunakan. Hasil input dari *user* akan disimpan pada variabel *metode_pencarian*.

d) Node 4

Node 4 pada notasi diagram alir mewakili simbol *decision* pada diagram alir aplikasi. *Node 4* mewakili proses percabangan

berdasarkan nilai variabel *metode_pencarian*. Jika variabel *metode_pencarian* bernilai *true* maka aliran proses akan menuju *node 5* sedangkan jika variabel *metode_pencarian* bernilai *false* maka aliran proses akan menuju *node 11*.

e) Node 5

Node 5 pada notasi diagram alir mewakili simbol manual input pada diagram alir aplikasi. *Node 5* akan dilalui ketika *user* memilih metode pencarian berupa cari arti. *Node 5* mewakili proses input manual untuk mendapatkan data dari *user* terkait dengan kata kunci pencarian yang digunakan. Hasil input dari *user* akan disimpan pada variabel *input_pencarian*.

f) Node 6

Node 5 pada notasi diagram alir mewakili simbol *process* pada diagram alir aplikasi. *Node 6* mewakili proses pencarian data pada *database* berdasarkan kata kunci pencarian.

g) Node 7

Node 7 pada notasi diagram alir mewakili simbol *process* pada diagram alir aplikasi. *Node 7* mewakili proses mengisi variabel *hasil_pen-carian* berdasarkan hasil pencarian data pada *database*.

h) Node 8

Node 8 pada notasi diagram alir mewakili simbol *decision* pada diagram alir aplikasi. *Node 8* mewakili proses percabangan berdasarkan nilai variabel *hasil_pencarian*. Jika variabel *hasil_pencarian* bernilai *true* maka aliran proses akan menuju

node 9 sedangkan jika variabel *hasil_pencarian* bernilai *false* maka aliran proses akan menuju *node* 10.

i) *Node* 9

Node 9 pada notasi diagram alir mewakili simbol *display* pada diagram alir aplikasi. *Node* 9 akan dilalui jika nilai variabel *hasil_pencarian* adalah *null*. *Node* 9 mewakili proses berupa tampilan aplikasi yang menyatakan bahwa hasil pencarian data pada *database* tidak ditemukan.

j) *Node* 10

Node 10 pada notasi diagram alir mewakili simbol *display* pada diagram alir aplikasi. *Node* 10 akan dilalui jika nilai variabel *hasil_pencarian* tidak bernilai *null*. *Node* 10 mewakili proses berupa tampilan aplikasi yang menampilkan hasil pencarian.

k) *Node* 11

Node 11 pada notasi diagram alir mewakili simbol *process* pada diagram alir aplikasi. *Node* 11 mewakili proses pemilihan metode pencarian dengan berdasarkan cari nama.

l) *Node* 12

Node 12 pada notasi diagram alir mewakili simbol manual input pada diagram alir aplikasi. *Node* 12 akan dilalui ketika *user* memilih metode pencarian berupa cari nama. *Node* 12 mewakili proses input manual untuk mendapatkan data dari *user* terkait dengan kata kunci pencarian yang digunakan. Hasil input dari *user* akan disimpan pada variabel *input_pencarian*.

m) Node 13

Node 13 pada notasi diagram alir mewakili simbol *process* pada diagram alir aplikasi. *Node 13* mewakili proses pencarian data pada *database* berdasarkan kata kunci pencarian.

n) Node 14

Node 14 pada notasi diagram alir mewakili simbol *process* pada diagram alir aplikasi. *Node 14* mewakili proses mengisi variabel hasil pencarian berdasarkan hasil pencarian data pada *database*.

o) Node 15

Node 15 pada notasi diagram alir mewakili simbol *decision* pada diagram alir aplikasi. *Node 15* mewakili proses percabangan berdasarkan nilai variabel *hasil_pencarian*. Jika variabel *hasil_pencarian* bernilai *true* maka aliran proses akan menuju *node 16* sedangkan jika variabel *hasil_pencarian* bernilai *false* maka aliran proses akan menuju *node 17*.

p) Node 16

Node 16 pada notasi diagram alir mewakili simbol *display* pada diagram alir aplikasi. *Node 16* akan dilalui jika nilai variabel *hasil_pencarian* adalah *null*. *Node 16* mewakili proses berupa tampilan aplikasi yang menyatakan bahwa hasil pencarian data pada *database* tidak ditemukan.

q) Node 17

Node 17 pada notasi diagram alir mewakili simbol *display* pada diagram alir aplikasi. *Node 17* akan dilalui jika nilai variabel

hasil_pen-carian tidak bernilai *null*. *Node 17* mewakili proses berupa tampilan aplikasi yang menampilkan hasil pencarian.

r) *Node 18*

Node 18 pada notasi diagram alir mewakili simbol *terminator* pada diagram alir aplikasi. *Node 18* mewakili tahapan terakhir (*stop*) untuk proses pencarian pada aplikasi.

2) Menghitung *Cyclomatic Complexity*

Menurut Roger S.Pressman (2002:538), bila *cyclomatic complexity* digunakan dalam konteks metode pengujian *basis path*, maka nilai yang terhitung untuk *cyclomatic complexity* menentukan jumlah jalur independen dalam *basis set* suatu program. Persamaan *cyclomatic complexity* menurut Roger S.Pressman (2002:539) yaitu :

$$V(G) = E - N + 2$$

Keterangan :

$V(G)$ = Jumlah *cyclomatic complexity*.

E = Jumlah *edge* pada notasi diagram alir.

N = Jumlah *node* pada notasi diagram alir.

Berdasarkan hasil analisis terhadap notasi diagram alir aplikasi(lihat gambar 30), diketahui jumlah *edge* (E) yaitu 20 dan jumlah *node* (N) yaitu 18. Jumlah *cyclomatic complexity* adalah :

$$V(G) = 20 - 18 + 2$$

$$V(G) = 4$$

3) Menentukan *Test Case*

Hasil perhitungan terhadap *cyclomatic complexity* akan menentukan jumlah jalur independen *basis set* aplikasi. Hasil perhitungan *cyclomatic complexity* adalah 4(empat), Jadi jumlah jalur independen pada aplikasi yang dikembangkan peneliti untuk pengujian *white box* adalah 4(empat). Berdasarkan jumlah jalur independen tersebut maka dibuatlah *test case* untuk melakukan pengujian. *Test case* tersebut yaitu :

- a) Jalur = 1,2,3,4,5,6,7,8,9

Test Case I

Test case ini terjadi ketika *user* memilih metode pencarian cari arti. Kata kunci pencarian yang dimasukkan berupa nama perintah *command line* tertentu. Pada *test case* ini hasil pencarian kata kunci dalam *database* ditemukan

- b) Jalur = 1,2,3,4,5,6,7,16,9

Test Case II

Test case ini terjadi ketika *user* memilih metode pencarian cari arti. Kata kunci pencarian yang dimasukkan berupa nama perintah *command line* tertentu. Pada *test case* ini hasil pencarian kata kunci dalam *database* tidak ditemukan

- c) Jalur = 1,2,3,10,11,12,13,14,15,9

Test Case III

Test case ini terjadi ketika *user* memilih metode pencarian cari nama. Kata kunci pencarian yang dimasukkan berupa

kata kerja tertentu. Pada *test case* ini hasil pencarian kata kunci dalam *database* ditemukan.

d) Jalur = 1,2,3,10,11,12,13,14,17,9

Test Case IV

Test case ini terjadi ketika *user* memilih metode pencarian cari nama. Kata kunci pencarian yang dimasukkan berupa kata kerja tertentu. Pada *test case* ini hasil pencarian kata kunci dalam *database* tidak ditemukan

4) Pengujian *Test Case*

Pengujian setiap *test case* tersebut menggunakan aplikasi *android virtual device*. *Screenshot* pengujian terlampir. Hasil pengujian *test case* tersebut yaitu :

a) *Test Case I*

Pada pengujian *test case I* aplikasi mampu menampilkan hasil pencarian berdasarkan metode pencarian cari arti dengan tepat. Berdasarkan hal tersebut dapat dikatakan pengujian pada *test case I* tercapai.

b) *Test Case II*

Pada pengujian *test case II* *user* memasukkan kata kunci pencarian yang tidak tepat/hasil tidak diketahui. Respon yang ditunjukkan aplikasi adalah memunculkan kotak dialog yang menyatakan bahwa kata kunci yang dimasukkan tidak tepat. Berdasarkan hal tersebut dapat dikatakan pengujian pada *test case II* tercapai.

c) *Test Case III*

Pada pengujian *test case III* aplikasi mampu menampilkan hasil pencarian berdasarkan metode pencarian cari nama dengan tepat. Berdasarkan hal tersebut dapat dikatakan pengujian pada *test case III* tercapai.

d) *Test Case IV*

Pada pengujian *test case IV* user memasukkan kata kunci pencarian yang tidak tepat/hasil tidak diketahui. Respon yang ditunjukkan aplikasi adalah memunculkan kotak dialog yang menyatakan bahwa kata kunci yang dimasukkan tidak tepat. Berdasarkan hal tersebut dapat dikatakan pengujian pada *test case IV* tercapai.

Hasil pengujian *test case* secara lengkap ditunjukkan oleh tabel 29 dibawah ini.

Tabel 29. Hasil Pengujian Test Case Aplikasi

| No | Test Case | | | | | | | | | | Ketercapaian | |
|----|-----------|---|---|----|----|----|----|----|----|---|--------------|-------|
| | | | | | | | | | | | Ya | Tidak |
| 1. | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | | V | |
| 2. | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 16 | 9 | | V | |
| 3. | 1 | 2 | 3 | 10 | 11 | 12 | 13 | 14 | 15 | 9 | V | |
| 4. | 1 | 2 | 3 | 10 | 11 | 12 | 13 | 14 | 17 | 9 | V | |

Hasil pengujian *white box* (lihat tabel 29) menggunakan metode *basis path testing* menunjukan seluruh *test case* berhasil dieksekusi minimal satu kali. Berdasarkan hasil tersebut dapat disimpulkan bahwa aplikasi ini lolos pengujian *white box*.

b. Pengujian *Black Box*

Pengujian *black box* dilakukan dengan menguji perangkat lunak dari segi fungsionalitas perangkat lunak. Fungsionalitas perangkat lunak yang diuji sesuai dengan *use case* pada tahap desain. Peneliti membagi pengujian menjadi enam bagian. Setiap bagian diuji sesuai dengan skenario *use case* pada tahap desain.

Proses pengujian menggunakan *software* yang bernama *Android Virtual Device*. Hasil pengujian *black box* (lihat lampiran 1) yaitu :

Tabel 30. Hasil Pengujian Fungsionalitas Membuka Aplikasi

| Aksi Aktor | Reaksi Sistem | Hasil Pengujian |
|---------------------|--|-----------------|
| <i>Alur Dasar</i> | | |
| 1. Memulai aplikasi | | |
| | 2. Memanggil <i>file</i> Halaman-Utama.java untuk menampilkan menu utama aplikasi yang terdiri dari 4 (empat) tombol. 3. Melakukan inisiasi hubungan ke <i>database</i> . | Sesuai |

Hasil pengujian fungsionalitas Membuka Aplikasi (lihat tabel 30) menunjukkan bahwa ketika *user* membuka aplikasi, reaksi sistem yang ditunjukkan yaitu menampilkan menu utama serta berhasil melakukan inisiasi hubungan ke *database*. Berdasarkan hal tersebut dapat dikatakan bahwa hasil pengujian fungsionalitas membuka aplikasi adalah sesuai.

Tabel 31. Hasil Pengujian Fungsionalitas Menu Petunjuk

| Aksi Aktor | Reaksi Sistem | Hasil Pengujian |
|----------------------------|--|-----------------|
| <i>Alur Dasar</i> | | |
| 1. Menekan tombol Petunjuk | | |
| | 2. Memanggil <i>file</i> Halaman-Petunjuk.java untuk menampilkan gambar petunjuk pemakaian aplikasi. | Sesuai |

Hasil pengujian fungsionalitas Menu Petunjuk (lihat tabel 31) menunjukkan bahwa ketika *user* menekan tombol petunjuk, reaksi perangkat lunak yang ditunjukkan adalah menampilkan gambar-gambar petunjuk penggunaan perangkat lunak. Berdasarkan hal tersebut dapat dikatakan bahwa hasil pengujian fungsionalitas petunjuk adalah sesuai.

Tabel 32. Hasil Pengujian Fungsionalitas Keluar Aplikasi

| Aksi Aktor | Reaksi Sistem | Hasil Pengujian |
|--------------------------|--|-----------------|
| <i>Alur Dasar</i> | | |
| 3. Menekan tombol Keluar | | |
| | 4. Memanggil <i>method</i> keluar untuk menghentikan aplikasi. | Sesuai |

Hasil pengujian fungsionalitas Keluar Aplikasi (lihat tabel 32) menunjukkan bahwa ketika *user* menekan tombol keluar, reaksi sistem yang ditunjukkan adalah keluar dari aplikasi. Berdasarkan hal tersebut dapat dikatakan bahwa hasil pengujian fungsionalitas keluar aplikasi adalah sesuai.

Tabel 33. Hasil Pengujian Fungsionalitas Info Pengembang

| Aksi Aktor | Reaksi Sistem | Hasil Pengujian |
|---------------------------|---|-----------------|
| <i>Alur Dasar</i> | | |
| 3. Menekan tombol Tentang | | |
| | 4. Memanggil <i>file</i> Halaman-Tentang untuk menampilkan identitas pengembang aplikasi. | Sesuai |

Hasil pengujian fungsionalitas Info Pengembang seperti yang ditampilkan pada tabel 33 menunjukkan bahwa ketika *user* menekan tombol tentang, reaksi sistem yang ditunjukan adalah menampilkan identitas pengembang. Berdasarkan hal tersebut dapat dikatakan bahwa hasil pengujian fungsionalitas tentang adalah sesuai.

Tabel 34 . Hasil Pengujian Fungsionalitas Cari Arti

| Aksi Aktor | Reaksi Sistem | Hasil Pengujian |
|---|---|-----------------|
| <i>Alur Dasar</i> | | |
| 3. Menekan menu tab Cari Arti | | |
| | 4. Menampilkan menu tab Cari Arti | Sesuai |
| 5. Memasukkan nama perintah <i>command line</i> tertentu | | |
| | 6. Menampilkan <i>autosuggest</i> nama perintah <i>command line</i> . | Sesuai |
| 7. Menekan salah satu pilihan yang ditampilkan <i>autosuggest</i> . | | |

Tabel 35 . Hasil Pengujian Fungsionalitas Cari Arti(lanjutan)

| Aksi Aktor | Reaksi Sistem | Hasil Pengujian |
|---|--|-----------------|
| 8. Menekan tombol Cari | | |
| | 9. Mengecek basisdata perintah <i>command line</i> tertentu berdasarkan input. 10. Menampilkan data yang ditemukan pada <i>database</i> . | Sesuai |
| 11. Menekan pilihan data yang ditampilkan | | |
| | 12. Menampilkan detail keterangan mengenai data perintah <i>command line</i> sesuai pilihan | Sesuai |
| Alur Alternatif | | |
| 1. Menekan menu tab Cari Arti | | |
| | 2. Menampilkan menu tab Cari Arti | Sesuai |
| 3. Memasukkan nama perintah <i>command line</i> yang salah. | | |
| | 4. Tidak menampilkan <i>auto-suggest</i> nama perintah <i>command line</i> . | Sesuai |
| 5. Menekan tombol Cari | | |
| | 6. memunculkan pesan “Kata Kunci Pencarian Tidak Ditemukan” | Sesuai |

Hasil pengujian fungsionalitas Cari Arti seperti yang ditampilkan pada tabel 34 dan 35 menunjukkan bahwa reaksi sistem yang ditunjukkan berdasarkan aksi *user* baik alur dasar maupun alur alternatif sudah sesuai. Berdasarkan hal tersebut dapat dikatakan bahwa hasil pengujian fungsionalitas cari arti adalah sesuai.

Tabel 36. Hasil Pengujian Fungsionalitas Cari Nama

| Aksi Aktor | Reaksi Sistem | Hasil Pengujian |
|--|---|-----------------|
| <i>Alur Dasar</i> | | |
| 16. Menekan <i>menu tab</i> Cari Nama | | |
| | 17. Menampilkan <i>menu tab</i> Cari Nama | Sesuai |
| 18. Memasukkan input kata kerja dasar tertentu | | |
| | 19. Menampilkan <i>autosuggest</i> kata kerja dasar tertentu. | Sesuai |
| 20. Menekan salah satu pilihan yang ditampilkan <i>autosuggest</i> . | | |
| 21. Menekan tombol Cari | | |
| | 22. Mengecek dalam <i>database</i> kata kerja dasar tertentu berdasarkan input. 23. Menampilkan data yang ditemukan pada <i>database</i> . | Sesuai |
| 24. Menekan pilihan data yang ditampilkan | | |
| | 25. Menampilkan detail keterangan mengenai data perintah <i>command line</i> sesuai pilihan | Sesuai |

Tabel 37. Hasil Pengujian Fungsionalitas Cari Nama(lanjutan)

| Aksi Aktor | Reaksi Sistem | Hasil Pengujian |
|---|--|-----------------|
| <i>Alur Alternatif</i> | | |
| 1. Menekan <i>menu tab</i> Cari Nama | | |
| | 2. Menampilkan <i>menu tab</i> Cari Nama | Sesuai |
| 3. Memasukkan nama perintah <i>command line</i> yang salah. | | |
| | 4. Tidak menampilkan <i>auto-suggest</i> nama perintah <i>command line</i> . | Sesuai |
| 7. Menekan tombol Cari | | |
| | 8. memunculkan pesan "Kata Kunci Pencarian Tidak Ditemukan" | Sesuai |

Hasil pengujian fungsionalitas Cari Nama seperti yang ditampilkan pada tabel 36 dan 37 menunjukkan bahwa reaksi sistem yang ditunjukkan berdasarkan aksi *user* baik menu alur dasar maupun alur alternatif sudah sesuai. Berdasarkan hal tersebut dapat dikatakan bahwa hasil pengujian fungsionalitas cari nama adalah sesuai.

c. Pengujian *Alpha*

1) Pengujian Validasi Oleh Ahli Media Pendidikan

Pada pengujian validasi oleh ahli media pendidikan, peneliti meminta bantuan kepada dua orang ahli media pendidikan untuk menilai perangkat lunak yang dikembangkan. Penilaian dilakukan dengan menggunakan tabel spesifikasi perangkat lunak. Hasil pengujian validasi oleh ahli media pendidikan ditunjukkan oleh tabel 38, 39, 40, 41, 42,43, dan 44 dibawah ini.

Tabel 38. Hasil Uji Spesifikasi Membuka Aplikasi

| Aktivitas/Menu | Hasil Kerja yang Diharapkan | Ketercapaian | |
|------------------|--|--------------|-------|
| | | Ya | Tidak |
| Membuka Aplikasi | Tampil Halaman Menu Utama yang menampilkan 4(empat) tombol yaitu tombol pencarian, tombol petunjuk, tombol info pengembang, dan tombol keluar. | V | |

Hasil pengujian spesifikasi perangkat lunak yang dilakukan oleh dua ahli media pendidikan menunjukkan hasil yang sama pada aktifitas Membuka Aplikasi (lihat tabel 38) bahwa perangkat lunak mampu menampilkan halaman menu utama yang terdiri dari 4(empat) tombol yaitu tombol pencarian, tombol petunjuk, tombol info pengembang, dan tombol keluar. Berdasarkan hal tersebut dapat dikatakan bahwa perangkat lunak memenuhi spesifikasi dalam aktifitas membuka aplikasi.

Tabel 39. Hasil Uji Spesifikasi Menu Utama

| Aktivitas/Menu | Hasil Kerja yang Diharapkan | Ketercapaian | |
|----------------|---|--------------|-------|
| | | Ya | Tidak |
| Menu Utama | Menu Utama dapat terbuka ketika aplikasi pertama dijalankan. | V | |
| | Halaman Pencarian dapat terbuka ketika <i>user</i> menekan tombol "Pencarian". | V | |
| | Halaman Petunjuk dapat terbuka ketika <i>user</i> menekan tombol "Petunjuk". | V | |
| | Halaman Identitas Pengembang dapat terbuka ketika <i>user</i> menekan tombol "Info Pengembang". | V | |
| | <i>User</i> dapat keluar aplikasi ketika menekan tombol "Keluar". | V | |

Hasil pengujian spesifikasi perangkat lunak yang dilakukan oleh dua ahli media pendidikan menunjukkan hasil yang sama pada halaman Menu Utama (lihat tabel 39) bahwa perangkat lunak mampu melakukan aktifitas-aktifitas sesuai hasil yang diharapkan. Berdasarkan hal tersebut dapat dikatakan bahwa perangkat lunak memenuhi spesifikasi pada halaman menu utama.

Tabel 40. Hasil Uji Spesifikasi Menu Pencarian

| Aktivitas/Menu | Hasil Kerja yang Diharapkan | Ketercapaian | |
|----------------|--|--------------|-------|
| | | Ya | Tidak |
| Menu Pencarian | Halaman Pencarian berupa tampilan <i>menu tab</i> Cari Arti dan <i>menu tab</i> Cari Nama dapat terbuka ketika <i>user</i> menekan tombol “Pencarian”. | V | |
| | <i>User</i> dapat memilih dan masuk ke <i>menu tab</i> Cari Arti. | V | |
| | <i>User</i> dapat memilih dan masuk ke <i>menu tab</i> Cari Nama. | V | |

Hasil pengujian spesifikasi perangkat lunak yang dilakukan oleh dua ahli media pendidikan menunjukkan hasil yang sama pada halaman Menu Pencarian (lihat tabel 40) bahwa perangkat lunak mampu melakukan aktifitas-aktifitas seperti menampilkan *menu tab* Cari Arti dan *menu tab* Cari Nama, masuk ke *menu tab* Cari Arti, dan masuk ke *menu tab* Cari Nama. Berdasarkan hal tersebut dapat dikatakan bahwa perangkat lunak memenuhi spesifikasi pada halaman menu pencarian

.Tabel 41. Hasil Uji Spesifikasi Menu *Tab* Cari Arti

| Aktivitas/Menu | Hasil Kerja yang Diharapkan | Ketercapaian | |
|---------------------------|---|--------------|-------|
| | | Ya | Tidak |
| Menu <i>Tab</i> Cari Arti | User dapat kembali ke menu utama dengan menekan tombol “Home” pada halaman <i>menu tab</i> Cari Arti. | V | |
| | <i>Input text</i> pada <i>menu tab</i> Cari Arti dapat berfungsi dengan baik. | V | |
| | Fungsi <i>auto suggestion</i> berupa perintah <i>command line</i> pada <i>input text</i> berfungsi dengan baik. | V | |
| | User dapat menghilangkan tulisan pada <i>input text</i> dengan menekan tombol “Hapus”. | V | |
| | User dapat menampilkan hasil pencarian dengan menekan tombol “Cari”. | V | |
| | Hasil pencarian aplikasi sesuai dengan kata kunci pencarian. | V | |
| | Aplikasi akan memunculkan pesan ketika hasil pencarian tidak ditemukan. | V | |
| | User dapat menampilkan detail informasi perintah <i>command line</i> dengan menekan salah satu hasil pencarian. | V | |

Hasil pengujian spesifikasi perangkat lunak yang dilakukan oleh dua ahli media pendidikan menunjukan hasil yang sama pada halaman Menu *Tab* Cari Arti (lihat tabel 41) bahwa perangkat lunak mampu melakukan aktifitas-aktifitas sesuai hasil yang diharapkan. Berdasarkan hal tersebut dapat dikatakan bahwa perangkat lunak memenuhi spesifikasi pada halaman Menu *Tab* Cari Arti.

Tabel 42. Hasil Uji Spesifikasi Menu *Tab* Cari Nama

| Aktivitas/Menu | Hasil Kerja yang Diharapkan | Ketercapaian | |
|---------------------------|--|--------------|-------|
| | | Ya | Tidak |
| Menu <i>tab</i> Cari Nama | <i>User</i> dapat kembali ke menu utama dengan menekan tombol “ <i>Home</i> ” pada halaman <i>menu tab</i> Cari Nama. | V | |
| | <i>Input text</i> pada <i>menu tab</i> Cari Nama dapat berfungsi dengan baik. | V | |
| | Fungsi <i>auto suggestion</i> berupa perintah <i>command line</i> pada <i>input text</i> berfungsi dengan baik. | V | |
| | <i>User</i> dapat menghilangkan tulisan pada <i>input text</i> dengan menekan tombol “Hapus”. | V | |
| | <i>User</i> dapat menampilkan hasil pencarian dengan menekan tombol “Cari”. | V | |
| | Hasil pencarian aplikasi sesuai dengan kata kunci pencarian. | V | |
| | Aplikasi akan memunculkan pesan ketika hasil pencarian tidak ditemukan. | V | |
| | <i>User</i> dapat menampilkan detail informasi perintah <i>command line</i> dengan menekan salah satu hasil pencarian. | V | |

Hasil pengujian spesifikasi perangkat lunak yang dilakukan oleh dua ahli media pendidikan menunjukan hasil yang sama pada halaman Menu *Tab* Cari Nama (lihat tabel 42) bahwa perangkat lunak mampu melakukan aktifitas-aktifitas sesuai hasil yang diharapkan. Berdasarkan hal tersebut dapat dikatakan bahwa perangkat lunak memenuhi spesifikasi pada halaman Menu *Tab* Cari Nama

.Tabel 43. Hasil Uji Spesifikasi Menu Petunjuk

| Aktivitas/Menu | Hasil Kerja yang Diharapkan | Ketercapaian | |
|----------------|---|--------------|-------|
| | | Ya | Tidak |
| Menu Petunjuk | User dapat melihat halaman petunjuk penggunaan aplikasi ketika menekan tombol “Petunjuk” pada menu utama. | V | |
| | User dapat melihat petunjuk penggunaan aplikasi sebelumnya dengan menggeser layar ke kanan. | V | |
| | User dapat melihat petunjuk penggunaan aplikasi selanjutnya dengan menggeser layar ke kiri. | V | |
| | User dapat kembali ke menu utama dengan menekan tombol “Home” pada layar. | V | |

Hasil pengujian spesifikasi perangkat lunak yang dilakukan oleh dua ahli media pendidikan menunjukkan hasil yang sama pada halaman Menu Petunjuk (lihat tabel 43) bahwa perangkat lunak mampu melakukan aktifitas-aktifitas sesuai hasil yang diharapkan. Berdasarkan hal tersebut dapat dikatakan bahwa perangkat lunak memenuhi spesifikasi pada halaman Menu Petunjuk.

Tabel 44. Hasil Uji Spesifikasi Menu Info Pengembang

| Aktivitas/Menu | Hasil Kerja yang Diharapkan | Ketercapaian | |
|----------------------|--|--------------|-------|
| | | Ya | Tidak |
| Menu Info Pengembang | User dapat melihat informasi tentang identitas pengembang ketika menekan tombol “Info Pengembang”. | V | |
| | User dapat kembali ke menu utama dengan menekan tombol “Home” pada layar. | V | |

Hasil pengujian spesifikasi perangkat lunak yang dilakukan oleh dua ahli media pendidikan menunjukkan hasil yang sama pada halaman Menu Info Pengembang (lihat tabel 44) bahwa perangkat lunak mampu melakukan aktifitas-aktifitas seperti menampilkan halaman informasi pengembang dan kembali ke menu utama ketika menekan tombol “home”. Berdasarkan hal tersebut dapat dikatakan bahwa perangkat lunak memenuhi spesifikasi pada halaman Menu Info Pengembang.

Berdasarkan tabel hasil pengujian validasi oleh ahli media pendidikan diatas, seluruh ahli media menyatakan hasil yang sama yaitu seluruh spesifikasi yang diharapkan ada sudah sesuai dengan unjuk kerja perangkat lunak. Kedua penguji juga menyatakan bahwa perangkat lunak yang dikembangkan sudah memiliki unjuk kerja yang baik.

2) Pengujian Kualitas Perangkat Lunak dari segi *functionality*, *efficiency*, dan *usability*

Pengujian kualitas perangkat lunak dilakukan dalam rangka memvalidasi perangkat lunak yang dikembangkan. Pengujian kualitas perangkat lunak dilakukan dengan menggunakan instrument berupa kuesioner yang dikembangkan oleh peneliti. Instrument tersebut sudah dikembangkan berdasarkan tahap pengembangan instrument kuesioner menurut Jogiyanto sehingga hasil yang didapat nantinya dapat dinyatakan valid.

Pengujian kualitas perangkat lunak untuk aspek *functionality*, *efficiency*, dan *usability* menggunakan subjek *Independent Testing Group* yang terdiri dari 15 siswa SMK N 1 Bantul Kompetensi keahlian

Teknik Komputer dan Jaringan. Kuesioner yang digunakan sudah lolos uji reliabilitas (lihat lampiran 7) . Hasil pengujian aspek terhadap aspek *functionality*, *efficiency*, dan *usability* ditunjukkan dalam tabel 45. Pada tabel 45 tersebut angka 5 mewakili jawaban responden “Sangat Setuju”, angka 4 mewakili jawaban res-ponden “Setuju” dan angka 3 mewakili jawaban responden ragu-ragu.

Tabel 45. Hasil Instrumen Pengujian *Alpha* Aspek *Functionality*, *Efficiency*, dan *Usability*

| Pernyataan | Responden | | | | | | | | | | | | | | | Rata-rata tiap item |
|---------------|-----------|---|---|---|---|---|---|---|---|----|----|----|----|----|----|------------------------|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | |
| Functionality | | | | | | | | | | | | | | | | |
| 1 | 4 | 4 | 4 | 4 | 5 | 5 | 4 | 4 | 5 | 4 | 5 | 4 | 5 | 4 | 4 | 4,33 |
| 2 | 5 | 4 | 4 | 4 | 5 | 5 | 4 | 4 | 5 | 4 | 5 | 4 | 5 | 4 | 4 | 4,40 |
| 3 | 5 | 5 | 5 | 5 | 5 | 5 | 4 | 4 | 5 | 5 | 5 | 5 | 4 | 4 | 4 | 4,67 |
| 4 | 4 | 5 | 5 | 5 | 5 | 5 | 4 | 4 | 5 | 5 | 5 | 5 | 4 | 4 | 4 | 4,60 |
| 5 | 4 | 4 | 4 | 4 | 5 | 5 | 4 | 4 | 5 | 4 | 4 | 4 | 4 | 4 | 5 | 4,27 |
| 6 | 4 | 5 | 5 | 5 | 5 | 5 | 4 | 4 | 5 | 5 | 5 | 5 | 5 | 4 | 4 | 4,67 |
| 7 | 5 | 5 | 5 | 5 | 4 | 4 | 4 | 4 | 4 | 5 | 5 | 5 | 5 | 4 | 4 | 4,53 |
| Efficiency | | | | | | | | | | | | | | | | |
| 8 | 4 | 5 | 5 | 5 | 5 | 5 | 5 | 4 | 4 | 5 | 5 | 5 | 4 | 4 | 5 | 4,67 |
| 9 | 4 | 5 | 5 | 5 | 5 | 5 | 5 | 4 | 4 | 5 | 5 | 5 | 4 | 4 | 5 | 4,67 |
| 10 | 5 | 5 | 5 | 5 | 4 | 4 | 5 | 4 | 4 | 4 | 4 | 4 | 5 | 3 | 4 | 4,33 |
| Usability | | | | | | | | | | | | | | | | |
| 11 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 5 | 5 | 5 | 4 | 4 | 4 | 4,20 |
| 12 | 4 | 4 | 5 | 4 | 4 | 4 | 4 | 4 | 5 | 5 | 5 | 5 | 5 | 4 | 5 | 4,47 |
| 13 | 5 | 4 | 5 | 4 | 5 | 5 | 4 | 4 | 5 | 5 | 5 | 5 | 4 | 4 | 5 | 4,60 |
| 14 | 3 | 5 | 5 | 5 | 5 | 5 | 5 | 4 | 5 | 3 | 3 | 3 | 3 | 4 | 4 | 4,13 |
| 15 | 4 | 5 | 5 | 5 | 5 | 5 | 4 | 4 | 4 | 5 | 5 | 5 | 4 | 4 | 5 | 4,60 |
| 16 | 4 | 4 | 4 | 4 | 5 | 5 | 4 | 4 | 5 | 5 | 5 | 5 | 4 | 4 | 4 | 4,40 |
| 17 | 4 | 5 | 5 | 5 | 5 | 5 | 4 | 4 | 4 | 4 | 3 | 4 | 4 | 4 | 4 | 4,27 |
| 18 | 3 | 5 | 5 | 5 | 4 | 5 | 4 | 4 | 3 | 4 | 3 | 4 | 4 | 3 | 4 | 4,00 |

Pengujian kualitas perangkat lunak untuk aspek *portability* dilakukan oleh pengembang menggunakan aplikasi *android virtual device*. *Screenshot* pengujian *portability* terlampir. Hasil pengujian *portability* (lihat lampiran 2)perangkat aplikasi ditunjukkan dalam tabel 46 dibawah ini.

Tabel 46. Hasil Pengujian Kualitas Perangkat Lunak Aspek *Portability*

| Variabel | Indikator | Sub Indikator | Hasil Yang diharapkan | Ketercapaian | |
|-------------|----------------|--|---|--------------|-------|
| | | | | Ya | Tidak |
| Portability | Installability | Keberhasilan dalam instalasi | Aplikasi <i>FyComm</i> berhasil diinstal pada android versi <i>Gingerbread</i> | V | |
| | | | Aplikasi <i>FyComm</i> berhasil diinstal pada android versi <i>Ice Cream Sandwich</i> | V | |
| | | | Aplikasi <i>FyComm</i> berhasil diinstal pada android versi <i>Jelly Bean</i> | V | |
| | Adaptability | Penyesuaian terhadap jenis kerapatan layar | Aplikasi <i>FyComm</i> dapat dijalankan pada resolusi layar 480x800 dengan 240 dpi | V | |
| | | | Aplikasi <i>FyComm</i> dapat dijalankan pada resolusi layar 320x480 dengan 160 dpi | V | |
| | | | Aplikasi <i>FyComm</i> dapat dijalankan pada resolusi layar 240x320 dengan 120 dpi | V | |
| | | Penyesuaian terhadap orientasi layar | Aplikasi dapat dijalankan pada orientasi <i>portrait</i> | V | |
| | | | Aplikasi dapat dijalankan pada orientasi <i>landscape</i> | V | |

Hasil pengujian aspek portabilitas (lihat tabel 46) perangkat lunak menunjukkan bahwa ketika aplikasi diinstall pada tiga sistem android yang berbeda yaitu *gingerbread*, *ice cream sandwich*, dan *jelly bean* aplikasi dapat berjalan dengan sempurna. Perangkat lunak juga dapat berjalan dengan baik ketika diterapkan pada kerapatan piksel 480x800, 320x320,

dan 240x320. Pada saat diterapkan pada orientasi *portrait* dan *landscape* perangkat lunak juga dapat berjalan dengan baik.

a. Uji Materi

Uji Materi dilakukan untuk melakukan validasi terhadap materi yang ditampilkan aplikasi. Uji materi dilakukan dengan memberikan kuesioner berisi materi kepada ahli. Ahli materi yang dipilih adalah dua orang pengajar Kompetensi keahlian Teknik Komputer dan Jaringan di SMK N 1 Bantul yaitu Bapak Wakhid Bashori, S.Pd dan Ibu Diah Utaminingsih, S.T. Hasil pengujian materi terlampir. Hasil uji materi (lihat lampiran 6) yang dilakukan oleh dua ahli tersebut menyatakan bahwa materi yang ditampilkan oleh aplikasi ini valid.

B. Pembahasan Hasil Pengujian Pengujian Kualitas Perangkat Lunak

Hasil pengujian kualitas perangkat lunak yang ditunjukkan oleh tabel 45 dan 46 akan dianalisis untuk mendapatkan kesimpulan terkait ketercapaian aspek kualitas perangkat lunak. Perhitungan yang dilakukan untuk menganalisis hasil pengujian yaitu :

a. Aspek *Functionality*, *Efficiency*, dan *Usability*

Perhitungan untuk proses analisis hasil pengujian *alpha* aspek *functionality*, *efficiency*, dan *usability* ditunjukkan oleh tabel 47 dibawah ini.

Tabel 47. Perhitungan Hasil Pengujian *Alpha* Aspek *Functionality*, *Efficiency*, dan *Usability*.

| Responden | Functionality | Efficiency | Usability |
|---------------|---------------|------------|-----------|
| 1 | 31 | 13 | 31 |
| 2 | 32 | 15 | 36 |
| 3 | 32 | 15 | 38 |
| 4 | 32 | 15 | 36 |
| 5 | 34 | 14 | 37 |
| 6 | 34 | 14 | 38 |
| 7 | 28 | 15 | 33 |
| 8 | 28 | 12 | 32 |
| 9 | 34 | 12 | 35 |
| 10 | 32 | 14 | 36 |
| 11 | 34 | 14 | 34 |
| 12 | 32 | 14 | 36 |
| 13 | 32 | 13 | 32 |
| 14 | 28 | 11 | 31 |
| 15 | 29 | 14 | 35 |
| Total Skor | 472 | 205 | 520 |
| Skor Maksimum | 525 | 225 | 600 |
| Persentase | 0.899 | 0.9111 | 0.8666 |

b. Aspek *Portability*

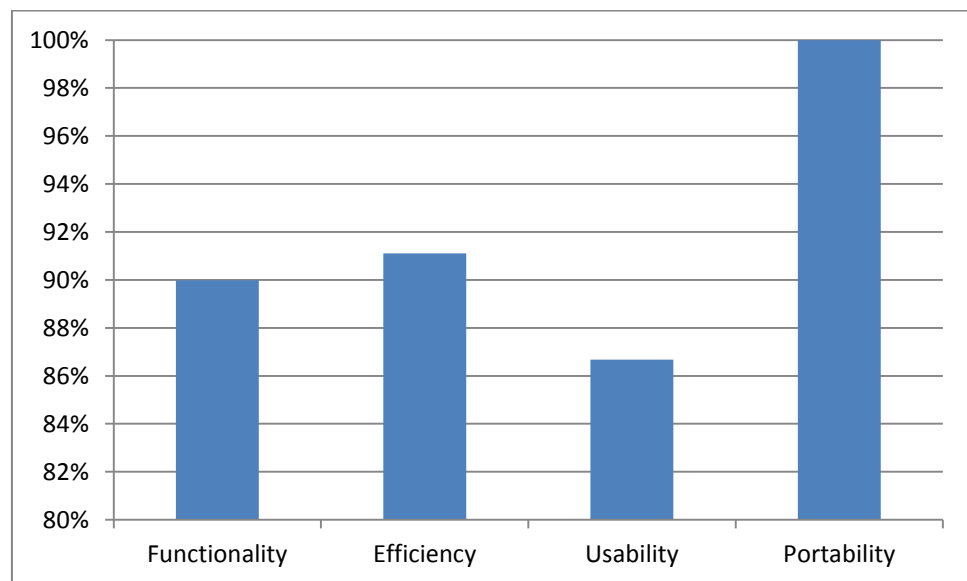
Hasil pengujian *alpha* untuk aspek *portability* (lihat tabel 46) menunjukkan bahwa seluruh pernyataan terpenuhi. Berdasarkan hal tersebut dapat disimpulkan bahwa persentase kelayakan aspek *portability* sebesar 100%.

Hasil-hasil perhitungan yang sudah dilakukan tersebut digunakan untuk mengetahui ketercapaian perangkat lunak terhadap kualitas perangkat lunak dari aspek *functionality*, *efficiency*, *usability* dan *portability*. Ketercapaian kualitas perangkat lunak yang dikembangkan terhadap aspek *functionality*, *efficiency*, *usability*, dan *portability* ditunjukkan oleh tabel 48 di bawah ini.

Tabel 48. Tingkat Kelayakan Perangkat Lunak dari Aspek *Functionality, Efficiency, dan Usability*

| No | Faktor | Persentase | Tingkat Kelayakan |
|----|---------------|------------|-------------------|
| 1 | Functionality | 89.9 % | Sangat Layak |
| 2 | Efficiency | 91.11 % | Sangat Layak |
| 3 | Usability | 86.67 % | Sangat Layak |
| 4 | Portability | 100 % | Sangat Layak |

Berdasarkan tabel 48 tersebut dapat diketahui bahwa keseluruhan faktor kualitas perangkat lunak telah dipenuhi dengan predikat “sangat layak”. Hasil perhitungan kuesioner pengujian *aspek functionality, efficiency, usability, dan portability* dapat digambarkan dengan diagram persentase. Diagram hasil perhitungan persentase hasil pengujian *alpha* aspek *functionality, efficiency, usability, dan portability* sebagai berikut ini :



Gambar 31. Diagram Persentase Hasil Pengujian *Alpha* Aspek *Functionality, Efficiency, Usability, dan Portability*.

C. Hasil Akhir Produk

Perangkat lunak Aplikasi *Android* Kamus *Command Line* ini telah dikembangkan sesuai dengan pedoman *System Development Life Cycle*. Tahapan-tahapan yang sudah dilalui yaitu analisis kebutuhan, desain,

implementasi, dan pengujian. Tahapan-tahapan tersebut sesuai dengan model *waterfall*. Tahapan analisis kebutuhan menghasilkan spesifikasi-spesifikasi yang harus dimiliki oleh perangkat lunak. Hasil tahapan analisis kebutuhan tersebut kemudian diproses ditahapan desain. Hasil dari tahapan desain adalah *use case*, diagram *sequence*, dan diagram alir aplikasi. Tahapan implementasi dilakukan berdasarkan hasil tahapan desain yang sudah dilakukan.

Tahapan akhir pengembangan adalah pengujian. Pengujian dilakukan dalam rangka melakukan verifikasi dan validasi perangkat lunak. Proses verifikasi dilakukan dengan melakukan pengujian *white box* sedangkan proses validasi dilakukan dengan melakukan pengujian *black box* dan *pengujian alpha*. Pengujian lain terkait validasi perangkat lunak yaitu pengujian materi. Uji materi dilakukan untuk memvalidasi materi berupa perintah *command line* yang ditampilkan aplikasi.

Hasil akhir dari penelitian ini adalah sebuah aplikasi *android* kamus *command line*. Perangkat tersebut memiliki dua fitur utama yang spesifik. Fitur-fitur tersebut yaitu proses pencarian berdasarkan nama perintah *command line* dan proses pencarian berdasarkan kata kerja tertentu.

BAB V

SIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan hasil yang didapat dari penelitian Pengembangan Aplikasi *Android* Kamus *Command Line* sebagai Media Bantu Belajar Siswa SMK Negeri 1 Bantul Kompetensi keahlian Teknik Komputer dan Jaringan maka peneliti mengambil kesimpulan yaitu :

1. Hasil dari Pengembangan Aplikasi *Android* Kamus *Command Line* sebagai Media Bantu Belajar Siswa SMK Negeri 1 Bantul Kompetensi keahlian Teknik Komputer dan Jaringan telah sesuai dengan spesifikasi yang diharapkan. Hasil perancangan ini didukung dari hasil pengujian alpha perangkat lunak yang dilakukan oleh ahli rekayasa perangkat lunak dan disimpulkan bahwa perangkat dapat bekerja secara baik dan sesuai dengan spesifikasi yang diharapkan.
2. Hasil dari kualitas perangkat lunak hasil penelitian Pengembangan Aplikasi *Android* Kamus *Command Line* sebagai Media Bantu Belajar Siswa SMK Negeri 1 Bantul Kompetensi keahlian Teknik Komputer dan Jaringan masuk dalam kategori “Sangat Layak”. Hasil pengujian unjuk kerja ini didukung oleh hasil pengujian alpha perangkat lunak untuk setiap faktor yaitu : *functionality* sebesar 89.9%(sangat layak), *efficiency* sebesar 91.11%(sangat layak), *usability* sebesar 86.67%(sangat layak), dan *portability* sebesar 100%(sangat layak).

B. Saran

Pengembangan aplikasi yang dilakukan peneliti tentu masih terdapat banyak kekurangan. Peneliti memiliki pemikiran dan saran untuk pengembangan kedepannya antara lain :

1. Jumlah perintah *command line* yang ada dalam *database* lebih diperbanyak. Pada perangkat lunak yang dikembangkan oleh peneliti, daftar perintah *command line* yang ditampilkan hanya yang berhubungan dengan jaringan. Hal tersebut dikarenakan sasaran pengguna aplikasi adalah siswa kompetensi keahlian Teknik Komputer dan Jaringan. Penambahan jumlah perintah *command line* akan menjadikan perangkat lunak yang dikembangkan lebih bisa dimanfaatkan tidak hanya untuk siswa kompetensi keahlian Teknik Komputer dan Jaringan tetapi kalangan umum.
2. Penelitian yang dilakukan peneliti masih berfokus pada proses pengembangan dan pengujian. Penelitian ini belum meneliti tentang seberapa efektifnya aplikasi *FYComm* untuk meningkatkan hasil belajar siswa. Berkaitan dengan hal ini perlu adanya penelitian lanjut dikemudian hari untuk mengetahui tingkat efektifitas aplikasi *FYComm* untuk meningkatkan hasil belajar siswa.

DAFTAR PUSTAKA

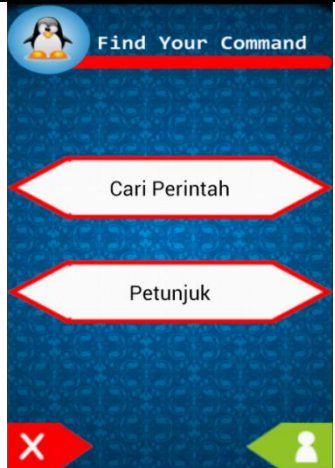
- Abdul Chaer.(2007). *Linguistik Umum*.Jakarta:PT.Rineka Cipta.
- Agarwall, B.B, et al.(2010). *Software Engineering & Testing*. London : Jones & Bartlett Publisher,LLC.
- Anonym.(2014).*Desktop Operating System Market Share*. Diakses dari <http://www.netmarket-share.com/operating-system-market-share.aspx?qprid=10&qp-customd=0&qptimeframe=Y> pada tanggal 19 Maret 2014 jam 22.00 WIB
- Bassil, Youssef.(2011). *A Simulation Model for the Waterfall Software Development Life Cycle*. International Journal of Engineering & Technology.
- David, Assaf Ben.(2011). *Mobile Application Testing*. Amdocs.
- Dodit Supriyanto & Rini Agustina.(2012). *Pemrograman Aplikasi Android*. Yogyakarta: MediaKom
- Harianti.(2013). *14 Juta Smartphone Terjual, Indonesia Negara Paling Konsumtif se-Asia Tenggara Menurut GfK Asia*.diakses dari <http://harianti.com/14-juta-smartphone-terjual-indonesia-negara-paling-konsumtif-se-asia-tenggara-menurut-gfk-asia/> pada tanggal 26 Maret 2014 jam 11.46 WIB.
- Hetzel, Bill.(1998).*The Complete Guide to Software Testing, Second Edition*. New York : John Wiley
- ISO/EIC 9126.(1996). *Information Technology – Software Product Evaluation – Quality Characteristics and Guidelines for Their Use*. International Standard
- Janner Simamarta.(2006).*Aplikasi Mobile Commerce menggunakan PHP dan MySQL*. Yogyakarta: Andi Offset.
- JD. Latuheru.(1988). *Media Pembelajaran dalam Proses Belajar Masa Kini*. Jakarta Depdikbud Dirjen PT. Proyek Pengembangan LPTK.
- Jogiyanto.(2008). *Metodologi Penelitian Sistem Informasi*.Yogyakarta:Andi
- Khan, Mohd.Ehmer.(2011). *Different Approach to White Box Testing Technique for Finding Errors*. International Journal of Software Engineering and Its Application
- Nidhra, Srinivas & Dondeti, Jagruthi.(2012). *Black Box and White Box Testing Techniques-A Literature Review*. International Journal of Embedded Systems and Applications.

- Niknejad, Aida.(2011). *A Quality Evaluation of an Android Smartphone Application*. University of Gothenburg.Gothenburg, Sweden
- Pressman, Roger S.(2002). *Rekayasa Perangkat Lunak: Pendekatan Praktisi (Buku I)*. Penerjemah: LN. Harnaningrum.Yogyakarta: Andi
- Pressman, Roger S.(2002). *Rekayasa Perangkat Lunak: Pendekatan Praktisi (Buku II)*. Penerjemah: LN. Harnaningrum.Yogyakarta: Andi
- Pocatilu, Paul.(2008). *Testing Java ME Applications*. Revista Informatica Economica Journal
- Riduwan & Sunarto. (2012). *Pengantar Statistika untuk Penelitian Pendidikan, Sosial, Komunikasi, Ekonomi, dan Bisnis*. Bandung : Alfabeta
- Romi Satrio W.(2006). *Penilaian Media Pembelajaran*. Diakses dari <http://romi-satriawahono.net/2006/06/21/aspek-dan-kriteria-penilaian-media-pembelajaran/> tanggal 8 September 2014 jam 07.43 WIB.
- Rosa & M, Shalahuddin .(2011). *Rekayasa Perangkat Lunak(Terstruktur dan Berorientasi Objek*. Bandung: Modula
- Sommerville, Ian.(2003). *Software Engineering*. Penerjemah: Dra. Yuhliza Hanum, M.Eng. Jakarta : Erlangga
- Sudaryono, et al.(2013). *Pengembangan Instrumen Penelitian Pendidikan*. Yogyakarta: Graha Ilmu.
- Sugiyono.(2008). *Metode Penelitian Kuantitatif Kualitatif dan R&D*. Bandung: Alfabeta.
- Supriyanto, Aji. (2005). *Pengantar Teknologi Informasi*. Jakarta: Salemba Infotek.
- Veronica Cahyadi.(2003). *The Effect of Interactive Engagement Teaching Method to Student Understanding of Introductory Physics at the Faculty of Engineering, University of Surabaya, Indonesia*. Jurnal 1-9.
- Wijaya Kusumah.(2009). *Pengertian Media Pembelajaran*. Diakses dari <http://media-grafika.com/pengertian-media-pembelajaran> pada tanggal 8 September 2014, Jam 07.09 WIB.
- Yuni Sugiarti.(2013). *Analisis dan Perancangan UML*. Yogyakarta: Graha Ilmu

LAMPIRAN


Lampiran 1 Screenshot Pengujian *Black Box*

1. Use Case Memulai Aplikasi

| Aksi Aktor | Reaksi Sistem | Screenshot Hasil Pengujian |
|---------------------|---|---|
| <i>Alur Dasar</i> | | |
| 1. Memulai aplikasi | | |
| | 2. Memanggil <i>file</i> HalamanUtama.java untuk menampilkan menu utama terdiri dari 4 (empat) tombol. 3. Melakukan inisiasi hubungan ke <i>database</i> . |  |

2. Use Case Petunjuk

| Aksi Aktor | Reaksi Sistem | Hasil Pengujian |
|----------------------------|---------------|-----------------|
| <i>Alur Dasar</i> | | |
| 1. Menekan tombol Petunjuk | | |


| | | |
|--|---|---|
| | 2. Memanggil <i>file</i> HalamanPetunjuk.java untuk menampilkan gambar petunjuk pemakaian aplikasi. |  |
|--|---|---|

3. Use Case Keluar Aplikasi

| Aksi Aktor | Reaksi Sistem | Hasil Pengujian |
|--------------------------|--|-----------------|
| <i>Alur Dasar</i> | | |
| 1. Menekan tombol Keluar | | |
| | 2. Memanggil <i>method</i> keluar untuk menghentikan aplikasi. | Sesuai |

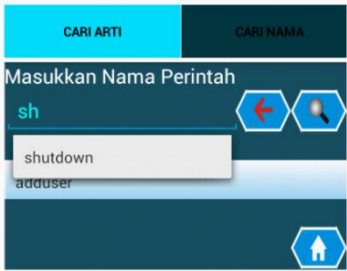

4. Use Case Info Pengembang




| Aksi Aktor | Reaksi Sistem | Hasil Pengujian |
|---------------------------|---------------|-----------------|
| <i>Alur Dasar</i> | | |
| 1. Menekan tombol Tentang | | |

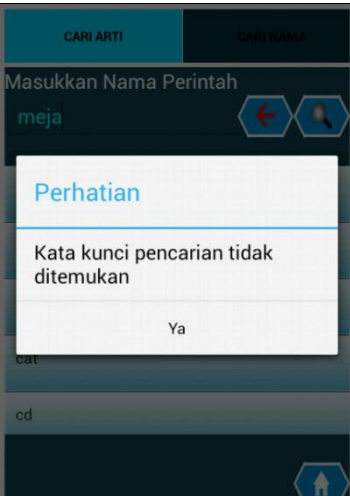
| | | |
|--|---|---|
| | 2. Memanggil <i>file</i> Halaman-Tentang untuk menampilkan identitas pengembang aplikasi. |  |
|--|---|---|

5. Use Case Cari Arti

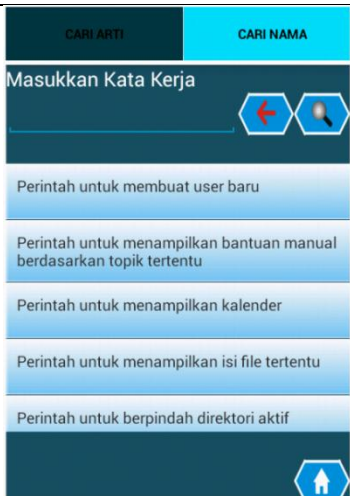
| Aksi Aktor | Reaksi Sistem | Hasil Pengujian |
|--|-----------------------------------|---|
| <i>Alur Dasar</i> | | |
| 1. Menekan menu tab Cari Arti | | |
| | 2. Menampilkan menu tab Cari Arti |  |
| 3. Memasukkan nama perintah <i>command line</i> tertentu | | |


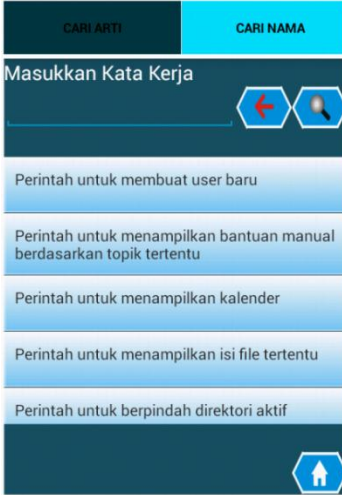
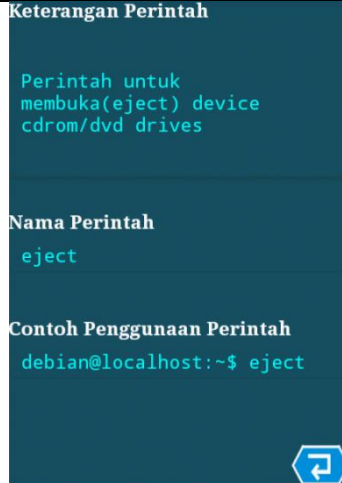
| | | |
|---|---|--|
| | 4. Menampilkan <i>auto-suggest</i> nama perintah <i>command line</i> . |  |
| 5. Menekan salah satu pilihan yang ditampilkan <i>autosuggest</i> . | | |
| 6. Menekan tombol Cari | | |
| | 7. Mengecek basisdata perintah <i>command line</i> tertentu berdasarkan input. 8. Menampilkan data yang ditemukan pada <i>database</i> . |  |
| 9. Menekan pilihan data yang ditampilkan | | |

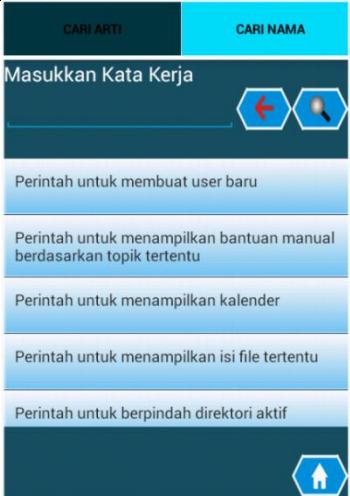
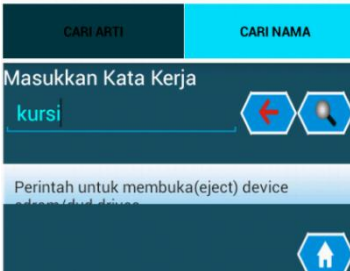
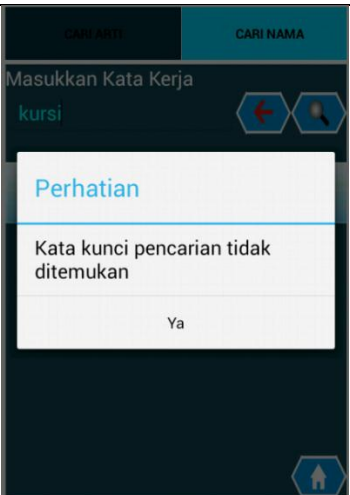
| | | |
|---|---|---|
| | 10. Menampilkan detail keterangan mengenai data perintah <i>command line</i> sesuai pilihan |  |
| <i>Alur Alternatif</i> | | |
| 1. Menekan menu tab Cari Arti | | |
| | 2. Menampilkan menu tab Cari Arti |  |
| 3. Memasukkan nama perintah <i>command line</i> yang salah. | | |
| | 4. Tidak menampilkan <i>auto suggest</i> nama perintah <i>command line</i> . |  |

| | | |
|------------------------|---|---|
| 5. Menekan tombol Cari | | |
| | 6. memunculkan pesan "Kata Kunci Pencarian Tidak Ditemukan" |  |

6. Use Case Cari Nama

| Aksi Aktor | Reaksi Sistem | Hasil Pengujian |
|--|--|---|
| <i>Alur Dasar</i> | | |
| 1. Menekan <i>menu tab</i> Cari Nama | | |
| | 2. Menampilkan <i>menu tab</i> Cari Nama |  |
| 3. Memasukkan input kata kerja dasar tertentu tertentu | | |

| | | |
|---|---|---|
| | 4. Menampilkan <i>auto suggest</i> kata kerja dasar tertentu. |  |
| 5. Menekan salah satu pilihan yang ditampilkan <i>autosuggest</i> . | | |
| 6. Menekan tombol Cari | | |
| | 7. Mengecek dalam <i>database</i> kata kerja dasar tertentu berdasarkan input. 8. Menampilkan data yang ditemukan pada <i>database</i> . |  |
| 9. Menekan pilihan data yang ditampilkan | | |
| | 10. Menampilkan detail keterangan mengenai data perintah <i>command line</i> sesuai pilihan |  |
| Alur Alternatif | | |



| | | |
|---|--|---|
| 1. Menekan <i>menu tab</i> Cari Nama | | |
| | 2. Menampilkan <i>menu tab</i> Cari Nama |  |
| 3. Memasukkan nama perintah <i>command line</i> yang salah. | | |
| | 4. Tidak menampilkan <i>auto-suggest</i> nama perintah <i>command line</i> . |  |
| 7. Menekan tombol Cari | | |
| | 8. memunculkan pesan "Kata Kunci Pencarian Tidak Ditemukan" |  |

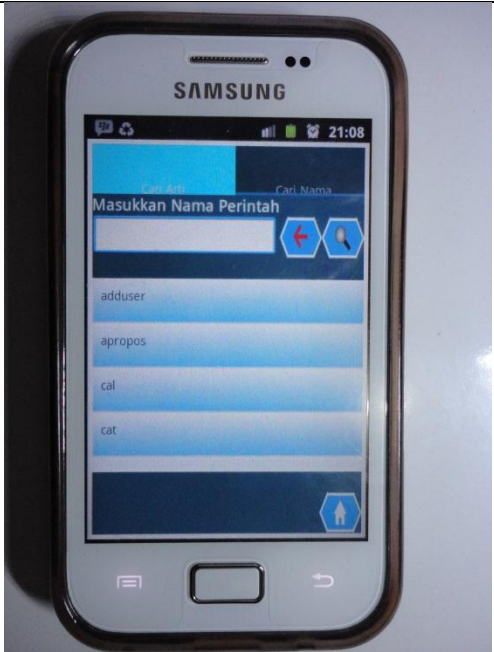
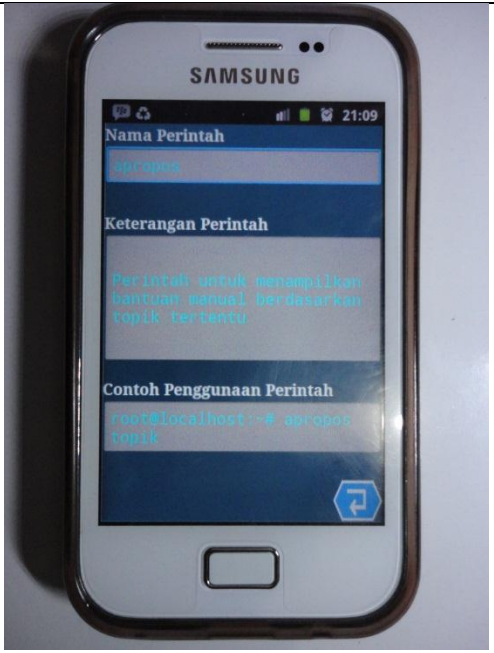
Lampiran 2 Screenshot Pengujian Aspek *Portability*


A. Hasil Pengujian Aspek *Portability* Sub Karakteristik *Installability*

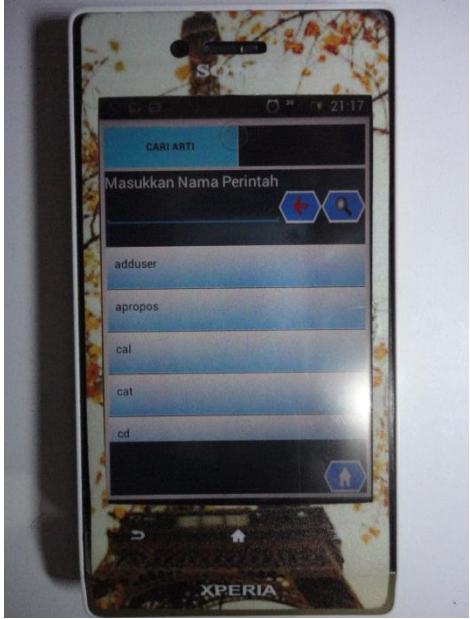

1. Keberhasilan dalam Instalasi

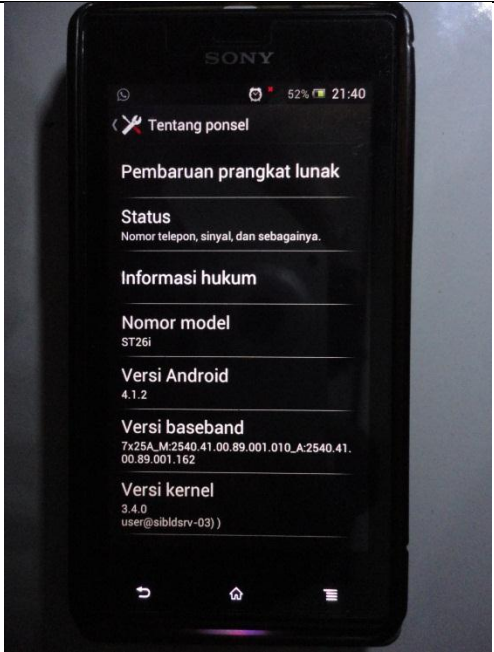
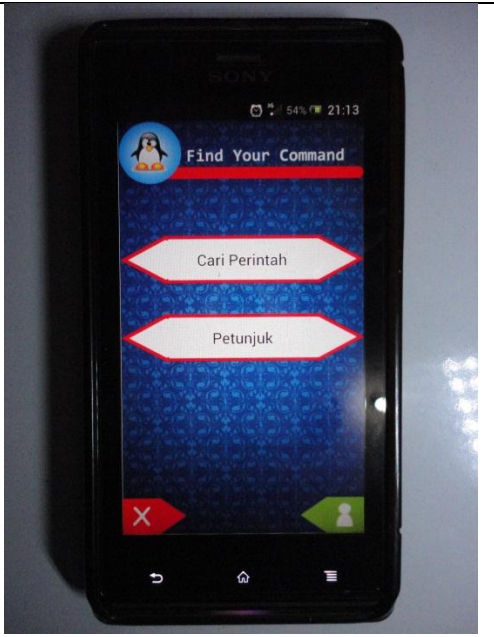
| No | <i>Device</i> untuk Pengujian | Sistem Operasi <i>Device</i> Pengujian | <i>Capture</i> Hasil Pengujian |
|----|-------------------------------------|---|--------------------------------|
|----|-------------------------------------|---|--------------------------------|

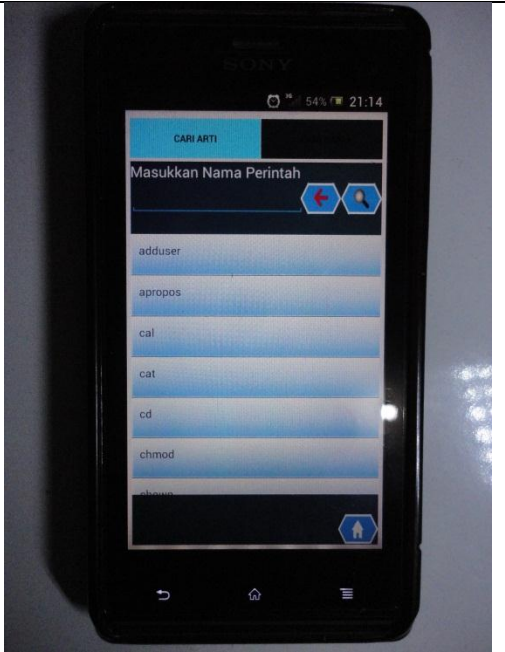
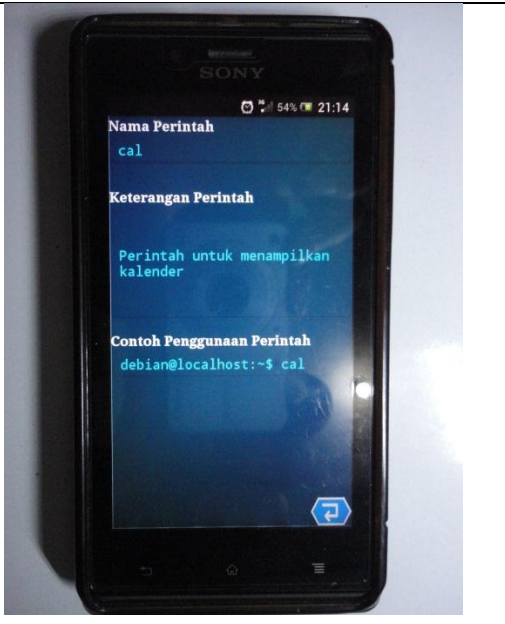
| | | | | |
|----|-------------------------------|----------------------|---|--|
| 1. | Samsung Galaxy Ace Plus | 2.3 (Gingerbread) | 1. Sistem Operasi yang Digunakan 2. Halaman Utama |   |
|----|-------------------------------|----------------------|---|--|

| | | | | |
|----|-------------------------------|----------------------|-----------------------------------|---|
| 1. | Samsung Galaxy Ace Plus | 2.3 (Gingerbread) | 3. Halaman Pencarian |  |
| | | | 4. Halaman Detail Informasi |  |

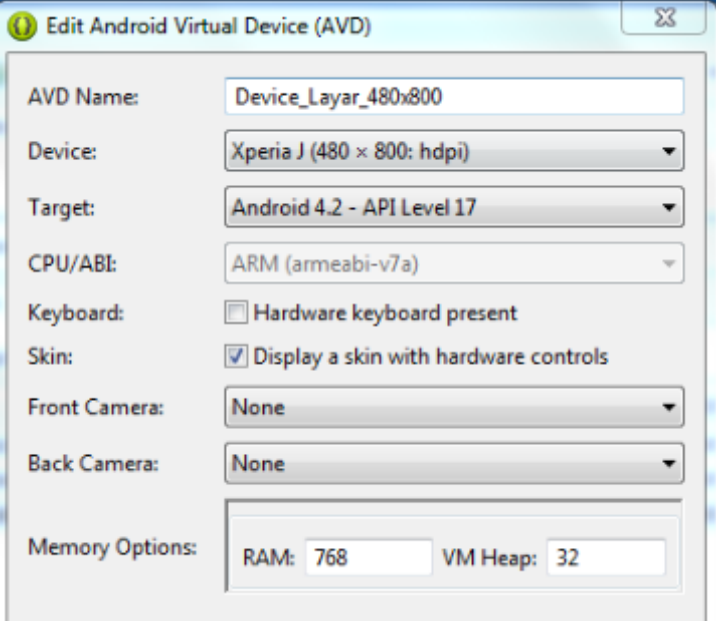
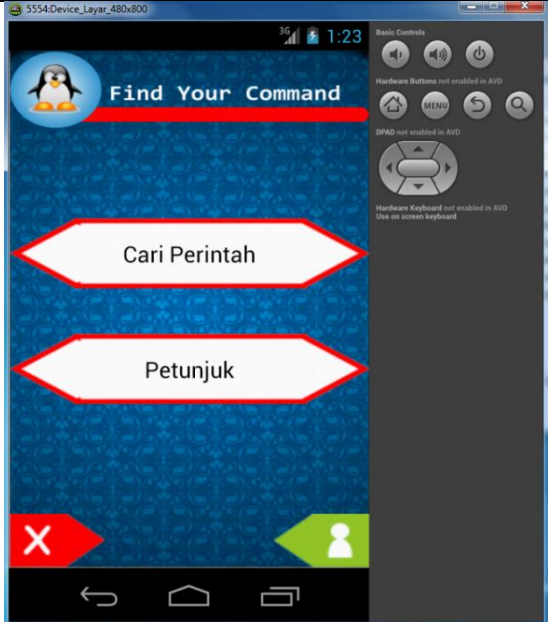
| | | | | |
|----|-------------------|--------------------------------|--|---|
| 2. | Sony Experia M | 4.0 (Ice Cream Sandwich) | 1. Sistem Operasi yang Digunakan |  |
| | | | 2. Halaman Utama |  |

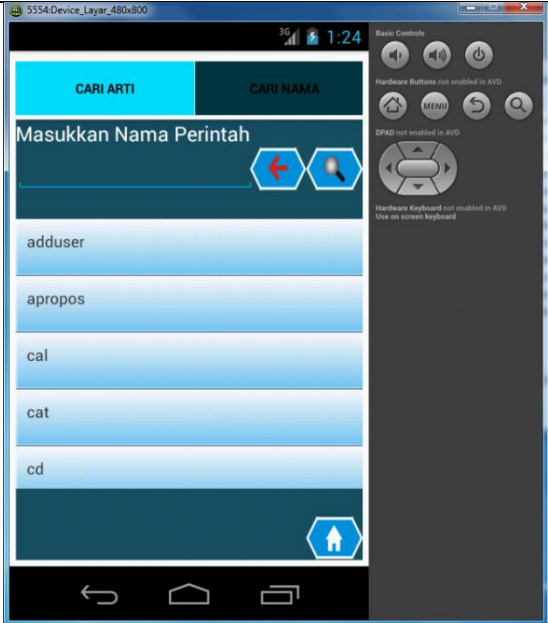

| | | | | |
|----|-------------------|--------------------------------|-----------------------------------|---|
| 2. | Sony Experia M | 4.0 (Ice Cream Sandwich) | 3. Halaman Pencarian |  |
| | | | 4. Halaman Detail Informasi |  |

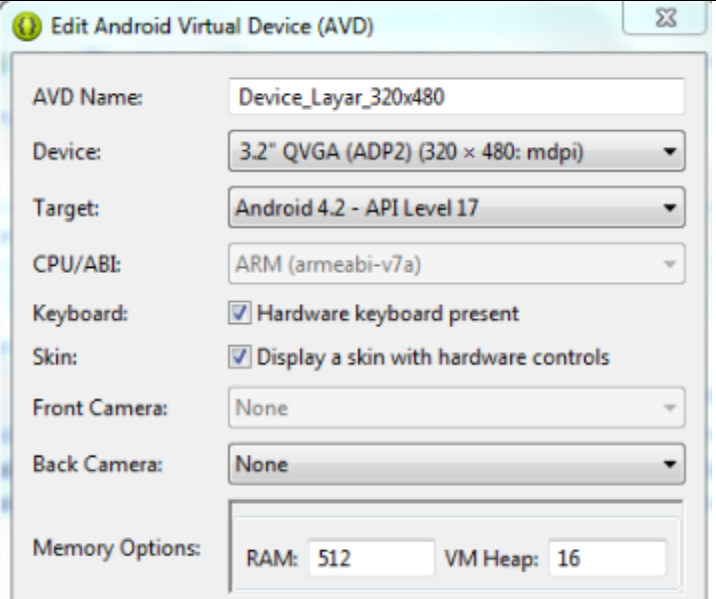

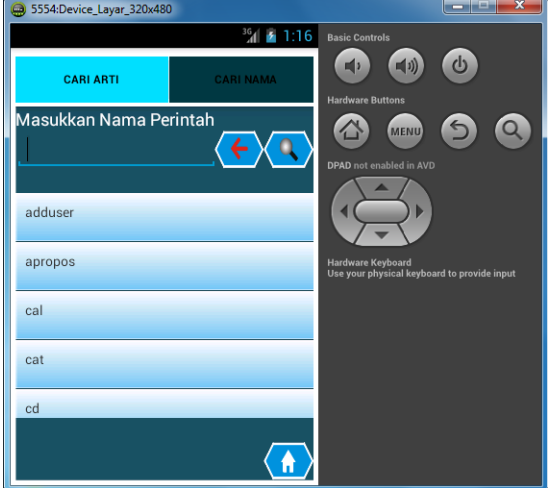
| | | | | |
|----|-------------------|---------------------|--|---|
| 3. | Sony Experia J | 4.1 (Jelly Bean) | 1. Sistem Operasi yang Digunakan |  |
| | | | 2. Halaman Utama |  |



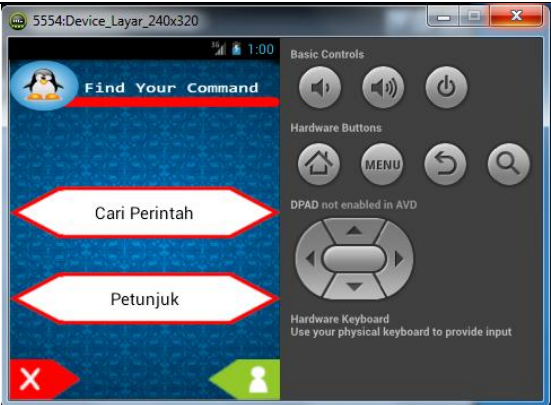
| | | | | |
|----|-------------------|---------------------|-----------------------------------|---|
| 3. | Sony Experia J | 4.1 (Jelly Bean) | 3. Halaman Pen- carian |  |
| | | | 4. Halaman Detail Informasi |  |


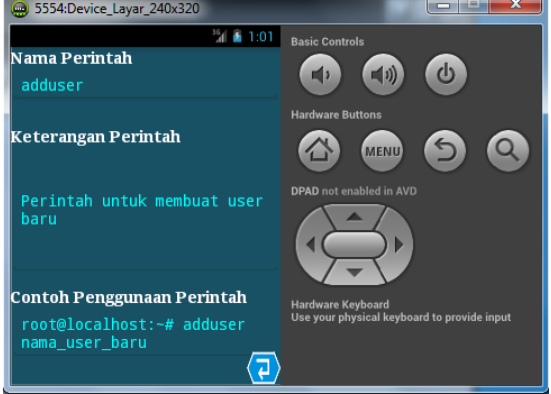
2. Penyesuaian Terhadap Jenis Kerapatan Layar

| No | Kerapatan Layar | Capture Hasil Pengujian | |
|----|-----------------|-------------------------------|--|
| 1. | 480 x 800 | 1. Spesifikasi Virtual Device |  |
| | | 2. Halaman utama |  |

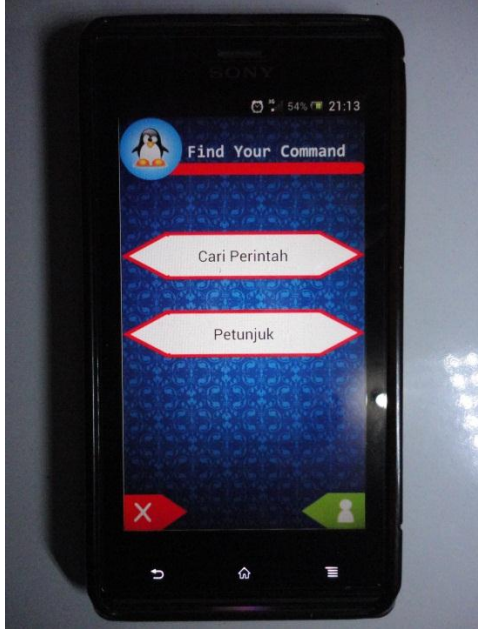
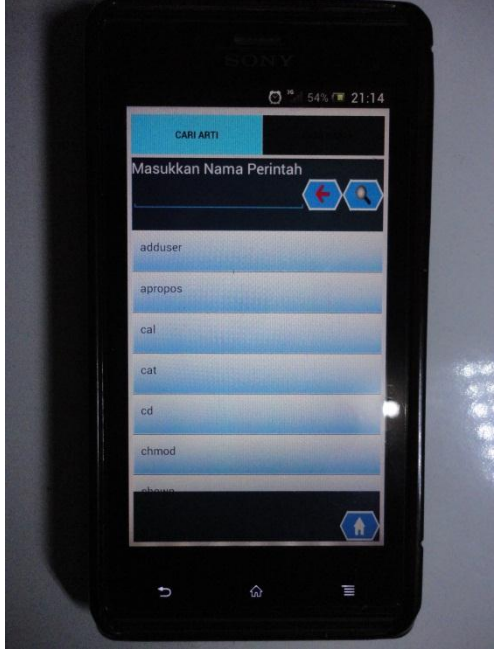
| | | | |
|----|-----------|-----------------------------|---|
| 1. | 480 x 800 | 3. Halaman Pencarian |  |
| | | 4. Halaman Detail Informasi |  |


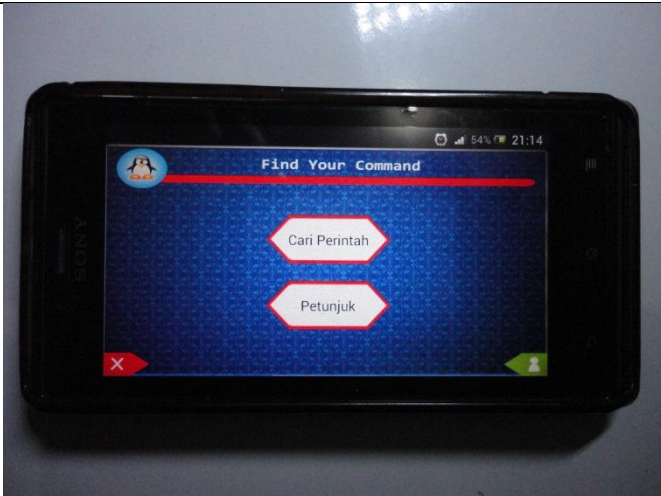
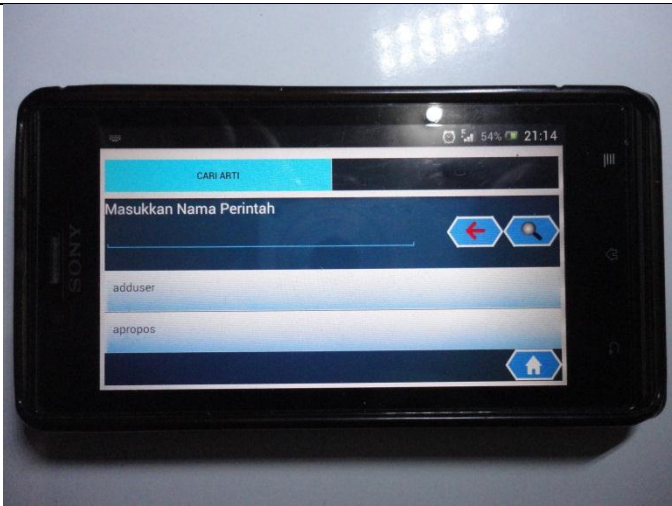
| | | | |
|----|-----------|-------------------------------|--|
| | | 1. Spesifikasi Virtual Device |  |
| 2. | 320 x 480 | 2. Halaman utama |  |
| | | 3. Halaman Pencarian |  |

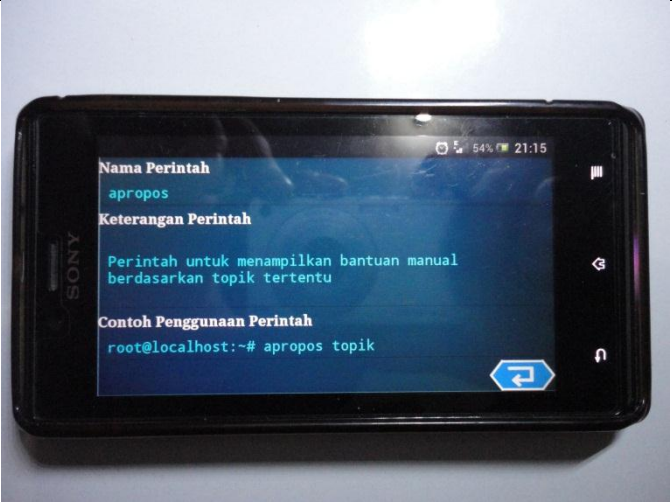
| | | | |
|----|-----------|-------------------------------|---|
| 2. | 320x480 | 4. Halaman Detail Informasi |  |
| 3. | 240 x 320 | 1. Spesifikasi Virtual Device |  |
| | | 2. Halaman utama |  |

| | | | |
|----|-----------|-----------------------------|---|
| 3. | 240 x 320 | 3. Halaman Pencarian |  |
| | | 4. Halaman Detail Informasi |  |

B. Hasil Pengujian Aspek *Portability* Sub Karakteristik *Adaptability*

| No | Orientasi Layar | Capture Hasil Pengujian | |
|----|-----------------|-------------------------|--|
| 1. | Portrait | 1. Halaman utama |  |
| | | 2. Halaman Pencarian |  |

| | | | |
|----|-----------|-----------------------------|--|
| 1. | Portrait | 3. Halaman Detail Informasi |  |
| 2. | Landscape | 1. Halaman utama |  |
| | | 2. Halaman Pencarian |  |

| | | | |
|----|-----------|-----------------------------|--|
| 2. | Landscape | 3. Halaman Detail Informasi |  |
|----|-----------|-----------------------------|--|

Lampiran 3 Validasi Instrumen Penelitian

SURAT PERNYATAAN VALIDASI INSTRUMEN PENELITIAN TUGAS AKHIR SKRIPSI

Saya yang bertanda tangan dibawah ini:

Nama : Drs. Suparman, M.Pd

NIP : 19550715 198003 1 006

Jurusan : Pendidikan Teknik Elektronika

menyatakan bahwa instrumen penelitian TAS atas nama mahasiswa:

Nama : Afied Hutomo

NIM : 10520241008

Program Studi : Pendidikan Teknik Informatika

Judul TAS : Pengembangan Aplikasi Android Kamus *Command Line*
sebagai Media Bantu Belajar Siswa SMK N 1 Bantul
Jurusan Teknik Komputer dan Jaringan.

Setelah dilakukan kajian atas instrumen penelitian TAS tersebut dapat
dinyatakan:

- ☒ Layak digunakan untuk penelitian
☐ Layak digunakan dengan perbaikan
☐ Tidak layak digunakan untuk penelitian yang bersangkutan

dengan saran/perbaikan sebagaimana terlampir.

Demikian agar dapat digunakan sebagaimana mestinya.

Yogyakarta, 11 Juni 2014

Validator,



Drs. Suparman, M.Pd

NIP. 19550715 198003 1 006

Catatan:

☐ Beri tanda ✓

INSTRUMEN PENELITIAN TUGAS AKHIR SKRIPSI

Saya yang bertanda tangan dibawah ini:

Nama : Dessy Irmawati, M.T
NIP : 19791214 201012 2 002
Jurusan : Pendidikan Teknik Elektronika

menyatakan bahwa instrumen penelitian TAS atas nama mahasiswa:

Nama : Afied Hutomo
NIM : 10520241008
Program Studi : Pendidikan Teknik Informatika
Judul TAS : Pengembangan Aplikasi *Android* Kamus *Command Line*
sebagai Media Bantu Belajar Siswa SMK N 1 Bantul
Jurusan Teknik Komputer dan Jaringan.

Setelah dilakukan kajian atas instrumen penelitian TAS tersebut dapat dinyatakan:

- ☒ Layak digunakan untuk penelitian
☐ Layak digunakan dengan perbaikan
☐ Tidak layak digunakan untuk penelitian yang bersangkutan

dengan saran/perbaikan sebagaimana terlampir.

Demikian agar dapat digunakan sebagaimana mestinya.

Yogyakarta, 13 Juni 2014

Validator,



Dessy Irmawati, M.T

NIP. 19791214 201012 2 002

Catatan:

☐ Beri tanda ✓

Lampiran 4 Hasil Pengujian *Alpha* pada Oleh Ahli Media Pendidikan

INSTRUMEN UJI SPESIFIKASI PERANGKAT LUNAK

E. Identitas Penguji

Nama = Ponco Wali Pranoto, M.Pd

Bidang Keahlian = Media Pendidikan

F. Petunjuk Pengisian

Berilah tanda *checklist* (✓) pada kolom pilihan "Ya" atau "Tidak" pada kolom ketercapaian yang ada pada tabel spesifikasi pengujian perangkat lunak.

G. Daftar Tabel Spesifikasi Pengujian

Tabel 1. Spesifikasi Uji Membuka Aplikasi

| Aktivitas/Menu | Hasil Kerja yang Diharapkan | Ketercapaian | |
|------------------|--|--------------|-------|
| | | Ya | Tidak |
| Membuka Aplikasi | Tampil Halaman Menu Utama yang menampilkan 4(empat) tombol yaitu tombol pencarian, tombol petunjuk, tombol info pengembang, dan tombol keluar. | ✓ | |

Tabel 2. Spesifikasi Uji Menu Utama

| Aktivitas/Menu | Hasil Kerja yang Diharapkan | Ketercapaian | |
|----------------|--|--------------|-------|
| | | Ya | Tidak |
| Menu Utama | Menu Utama dapat terbuka ketika aplikasi pertama dijalankan. | ✓ | |
| | Halaman Pencarian dapat terbuka ketika user menekan tombol "Pencarian". | ✓ | |
| | Halaman Petunjuk dapat terbuka ketika user menekan tombol "Petunjuk". | ✓ | |
| | Halaman Identitas Pengembang dapat terbuka ketika user menekan tombol "Info Pengembang". | ✓ | |
| | User dapat keluar aplikasi ketika menekan tombol "Keluar". | ✓ | |

Tabel 3. Spesifikasi Uji Menu Pencarian

| Aktivitas/Menu | Hasil Kerja yang Diharapkan | Ketercapaian | |
|----------------|--|--------------|-------|
| | | Ya | Tidak |
| Menu Pencarian | Halaman Pencarian berupa tampilan <i>menu tab</i> Cari Arti dan <i>menu tab</i> Cari Nama dapat terbuka ketika <i>user</i> menekan tombol "Pencarian". | ✓ | |
| | <i>User</i> dapat memilih dan masuk ke <i>menu tab</i> Cari Arti. | ✓ | |
| | <i>User</i> dapat memilih dan masuk ke <i>menu tab</i> Cari Nama. | ✓ | |

Tabel 4. Spesifikasi Uji Menu Tab Cari Arti

| Aktivitas/Menu | Hasil Kerja yang Diharapkan | Ketercapaian | |
|--------------------|--|--------------|-------|
| | | Ya | Tidak |
| Menu Tab Cari Arti | <i>User</i> dapat kembali ke menu utama dengan menekan tombol "Home" pada halaman <i>menu tab</i> Cari Arti. | ✓ | |
| | <i>Input text</i> pada <i>menu tab</i> Cari Arti dapat berfungsi dengan baik. | ✓ | |
| | Fungsi <i>auto suggestion</i> berupa perintah <i>command line</i> pada <i>input text</i> berfungsi dengan baik. | ✓ | |
| | <i>User</i> dapat menghilangkan tulisan pada <i>input text</i> dengan menekan tombol "Hapus". | ✓ | |
| | <i>User</i> dapat menampilkan hasil pencarian dengan menekan tombol "Cari". | ✓ | |
| | Hasil pencarian aplikasi sesuai dengan kata kunci pencarian. | ✓ | |
| | Aplikasi akan memunculkan pesan ketika hasil pencarian tidak ditemukan. | ✓ | |
| | <i>User</i> dapat menampilkan detail informasi perintah <i>command line</i> dengan menekan salah satu hasil pencarian. | ✓ | |

Tabel 5 . Spesifikasi Uji Menu Tab Cari Nama

| Aktivitas/Menu | Hasil Kerja yang Diharapkan | Ketercapaian | |
|--------------------|--|--------------|-------|
| | | Ya | Tidak |
| Menu tab Cari Nama | User dapat kembali ke menu utama dengan menekan tombol "Home" pada halaman menu tab Cari Nama. | ✓ | |
| | Input text pada menu tab Cari Nama dapat berfungsi dengan baik. | ✓ | |
| | Fungsi auto suggestion berupa kata kerja dasar tertentu pada input text berfungsi dengan baik. | ✓ | |
| | User dapat menghilangkan tulisan pada input text dengan menekan tombol "Hapus". | ✓ | |
| | User dapat menampilkan hasil pencarian dengan menekan tombol "Cari". | ✓ | |
| | Hasil pencarian aplikasi sesuai dengan kata kunci pencarian. | ✓ | |
| | Aplikasi akan memunculkan pesan ketika hasil pencarian tidak ditemukan. | ✓ | |
| | User dapat menampilkan detail informasi perintah command line dengan menekan salah satu hasil pencarian. | ✓ | |

Tabel 6. Spesifikasi Uji Menu Petunjuk

| Aktivitas/Menu | Hasil Kerja yang Diharapkan | Ketercapaian | |
|----------------|---|--------------|-------|
| | | Ya | Tidak |
| Menu Petunjuk | User dapat melihat halaman petunjuk penggunaan aplikasi ketika menekan tombol "Petunjuk" pada menu utama. | ✓ | |
| | User dapat melihat petunjuk penggunaan aplikasi sebelumnya dengan menggeser layar ke kanan. | ✓ | |
| | User dapat melihat petunjuk penggunaan aplikasi selanjutnya dengan menggeser layar ke kiri. | ✓ | |
| | User dapat kembali ke menu utama dengan menekan tombol "Home" pada layar. | ✓ | |

Tabel 5 . Spesifikasi Uji Menu Tab Cari Nama

| Aktivitas/Menu | Hasil Kerja yang Diharapkan | Ketercapaian | |
|--------------------|--|--------------|-------|
| | | Ya | Tidak |
| Menu tab Cari Nama | User dapat kembali ke menu utama dengan menekan tombol "Home" pada halaman menu tab Cari Nama. | ✓ | |
| | Input text pada menu tab Cari Nama dapat berfungsi dengan baik. | ✓ | |
| | Fungsi auto suggestion berupa kata kerja dasar tertentu pada input text berfungsi dengan baik. | ✓ | |
| | User dapat menghilangkan tulisan pada input text dengan menekan tombol "Hapus". | ✓ | |
| | User dapat menampilkan hasil pencarian dengan menekan tombol "Cari". | ✓ | |
| | Hasil pencarian aplikasi sesuai dengan kata kunci pencarian. | ✓ | |
| | Aplikasi akan memunculkan pesan ketika hasil pencarian tidak ditemukan. | ✓ | |
| | User dapat menampilkan detail informasi perintah command line dengan menekan salah satu hasil pencarian. | ✓ | |

Tabel 6. Spesifikasi Uji Menu Petunjuk

| Aktivitas/Menu | Hasil Kerja yang Diharapkan | Ketercapaian | |
|----------------|---|--------------|-------|
| | | Ya | Tidak |
| Menu Petunjuk | User dapat melihat halaman petunjuk penggunaan aplikasi ketika menekan tombol "Petunjuk" pada menu utama. | ✓ | |
| | User dapat melihat petunjuk penggunaan aplikasi sebelumnya dengan menggeser layar ke kanan. | ✓ | |
| | User dapat melihat petunjuk penggunaan aplikasi selanjutnya dengan menggeser layar ke kiri. | ✓ | |
| | User dapat kembali ke menu utama dengan menekan tombol "Home" pada layar. | ✓ | |

Tabel 7. Spesifikasi Uji Menu Info Pengembang

| Aktivitas/Menu | Hasil Kerja yang Diharapkan | Ketercapaian | |
|----------------------|--|--------------|-------|
| | | Ya | Tidak |
| Menu Info Pengembang | User dapat melihat informasi tentang identitas pengembang ketika menekan tombol "Info Pengembang". | ✓ | |
| | User dapat kembali ke menu utama dengan menekan tombol "Home" pada layar. | ✓ | |

H. Kesimpulan

Berdasarkan hasil dari tabel spesifikasi diatas, maka perangkat lunak hasil penelitian *Pengembangan Aplikasi Android Kamus Command Line sebagai Media Bantu Belajar Siswa SMK N 1 Bantul Jurusan Teknik Komputer dan Jaringan* in dinyatakan :

- ☒ Memiliki unjuk kerja yang baik
- ☐ Memiliki unjuk kerja kurang baik dan butuh revisi
- ☐ Tidak berfungsi

*beri tanda checklist pada salah satu pilihan.

I. Saran

Font & ukuran bab cari arti → cari nama diperbesar
warna lebih terang supaya mudah dibaca.

.....

.....

.....

.....

Yogyakarta, 3 Juli 2014

Penguji



Ponco Wali Pranoto, M.Pd

Lampiran 5 Hasil Pengujian *Alpha* oleh *Independent Testing Group*

UNTUK SISWA

INSTRUMEN PENELITIAN

Pengembangan Aplikasi *Android* Kamus *Command Line* sebagai Media Bantu Belajar

Siswa SMK Negeri 1 Bantul Jurusan Teknik Komputer dan Jaringan

A. Identitas Responden

Nama : Prilla Riana Dewi

Kelas : X TKJ 2

B. Petunjuk umum

1. Sebelum mengisi angket ini, pastikan Anda telah membaca dan menggunakan aplikasi *FyComm*.
2. Tulislah terlebih dahulu identitas Anda pada tempat yang sudah disediakan.
3. Bacalah dengan teliti setiap pernyataan dalam angket ini sebelum Anda memilih jawaban.
4. Jika ada yang tidak Anda mengerti, bertanyalah pada Guru atau Peneliti.
5. Petunjuk Penilaian
 - Isilah dengan memberi tanda (V) pada salah satu kolom jawaban penilaian sesuai dengan jawaban Anda.
 - Terdapat lima pilihan jawaban yang dapat dipilih yaitu sangat setuju (SS), setuju (S), netral (N), tidak setuju (TS), dan sangat tidak setuju (STS).
 - Jika pernyataan semakin sesuai, maka jawaban Anda semakin setuju.
 - Jika pernyataan semakin tidak sesuai, maka jawaban Anda semakin tidak setuju.
6. Atas kesediaan Anda untuk mengisi angket ini, saya ucapkan terimakasih.

A. Tabel Penilaian

| No | Pernyataan | Alternatif Jawaban | | | | |
|-------------------------------|--|--------------------|----|---|---|----|
| | | STS | TS | N | S | SS |
| Variabel <i>Functionality</i> | | | | | | |
| 1. | Perangkat lunak dapat melakukan pencarian berdasarkan metode pencarian Cari Nama. | | | | ✓ | |
| 2. | Perangkat lunak dapat melakukan pencarian berdasarkan metode pencarian Cari Arti. | | | | ✓ | |
| 3. | Perangkat lunak dapat menampilkan detail informasi perintah dari hasil metode pencarian Cari Nama. | | | | | ✓ |
| 4. | Perangkat lunak dapat menampilkan detail informasi perintah dari metode pencarian Cari Arti. | | | | | ✓ |
| 5. | Perangkat lunak dapat menampilkan halaman informasi pengembang aplikasi. | | | | ✓ | |
| 6. | Metode pencarian Cari Nama menghasilkan hasil yang sesuai dengan kata kunci yang dimasukkan. | | | | | ✓ |
| 7. | Metode pencarian Cari Arti menghasilkan hasil yang sesuai dengan kata kunci yang dimasukkan. | | | | | ✓ |
| Variabel <i>Efficiency</i> | | | | | | |
| 8. | Proses pencarian pada metode pencarian Cari Nama membutuhkan waktu yang singkat. | | | | | ✓ |
| 9. | Proses pencarian pada metode pencarian Cari Arti membutuhkan waktu yang singkat. | | | | | ✓ |
| 10. | Saya puas dengan durasi respon pada setiap fitur aplikasi. | | | | | ✓ |
| Variabel <i>Usability</i> | | | | | | |
| 11. | Aplikasi ini memberikan informasi dengan jelas. | | | | ✓ | |
| 12. | Informasi yang disediakan dapat membantu menyelesaikan tugas saya. | | | | ✓ | |
| 13. | Informasi yang disediakan aplikasi ini memudahkan saya dalam belajar. | | | | ✓ | |

| | | | | | | |
|-----|--|--|--|--|---|---|
| 14. | Interface aplikasi ini mudah dipelajari. | | | | | ✓ |
| 15. | Saya dapat mempelajari penggunaan aplikasi ini dengan mudah. | | | | | ✓ |
| 16. | Aplikasi ini menyediakan menu petunjuk penggunaan. | | | | ✓ | |
| 17. | Secara keseluruhan aplikasi ini nyaman ketika digunakan. | | | | | ✓ |
| 18. | Kemudahan dalam menggunakan aplikasi ini membuat saya puas. | | | | | ✓ |

B. Saran

lebih ditambahkan kalimat perimah-perimah,

.....

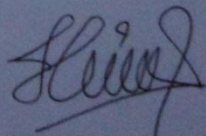
.....

.....

.....

Bantui, 16 Juni 2014

Penguji


(.....)

Lampiran 6 Hasil Pengujian Materi

SURAT KETERANGAN UJI MATERI

Saya yang bertanda tangan dibawah ini:

Nama : Diah Utaminingsih, S.T
NIP : 19770904 201101 2 004
Pekerjaan : Guru SMK N 1 Bantul Jurusan Teknik Komputer dan Jaringan.

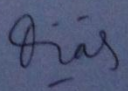
Setelah memeriksa uji materi dalam penelitian skripsi yang berjudul "Pengembangan Aplikasi *Android* Kamus *Command Line* sebagai Media Bantu Belajar Siswa SMK N 1 Bantul Jurusan Teknik Komputer dan Jaringan" dari mahasiswa Fakultas Teknik Universitas Negeri Yogyakarta atas nama :

Nama : Afied Hutomo
NIM : 10520241008
Program Studi : Pendidikan Teknik Informatika
Dosen Pembimbing : Dr. Eko Marpanaji

Materi berupa nama perintah *command line*, keterangan perintah *command line*, dan contoh penggunaan perintah *command line* dapat dinyatakan (Valid / Tidak Valid *)

Saran :
lebih Dipertanyakan lagi perintah-perintah yang dimasukkan dalam kamus (selengkap-lengkap nya)

Demikian agar dapat digunakan sebagaimana mestinya.

Bantul, Juni 2014
Validator,

Diah Utaminingsih, S.T
NIP. 19770904 201101 2 004

Catatan:
*) Lingkari pada salah satu pilihan yang sesuai dengan pendapat Ibu

SURAT KETERANGAN UJI MATERI

Saya yang bertanda tangan dibawah ini:

Nama : Wakhid Bashori, S.Pd
NIP : 19740104 200604 1 010
Pekerjaan : Guru SMK N 1 Bantul Jurusan Teknik Komputer dan Jaringan.

Setelah memeriksa uji materi dalam penelitian skripsi yang berjudul "Pengembangan Aplikasi *Android* Kamus *Command Line* sebagai Media Bantu Belajar Siswa SMK N 1 Bantul Jurusan Teknik Komputer dan Jaringan" dari mahasiswa Fakultas Teknik Universitas Negeri Yogyakarta atas nama :

Nama : Afied Hutomo
NIM : 10520241008
Program Studi : Pendidikan Teknik Informatika
Dosen Pembimbing : Dr. Eko Marpanaji

Materi berupa nama perintah *command line*, keterangan perintah *command line*, dan contoh penggunaan perintah *command line* dapat dinyatakan (Valid / Tidak Valid *)

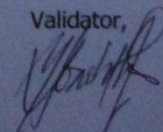
Saran :

.....
.....
.....
.....

Demikian agar dapat digunakan sebagaimana mestinya.

Bantul, Juni 2014

Validator,



Wakhid Bashori, S.Pd

NIP. 19740104 200604 1 010

Catatan:

*) Lingkari pada salah satu pilihan yang sesuai dengan pendapat Bapak

Lampiran 7 Hasil Uji Reliabilitas Instrumen

A. Hasil Pengujian Pilot

Berdasarkan hasil pengujian pilot yang dilakukan pada 10 responden, Menggunakan jumlah responden sebanyak 10 orang didasarkan pada pendapat Jogiyanto (2008:140). Hasil pengujian dapat dilihat pada tabel dibawah ini.

| Pernyataan | Responden | | | | | | | | | |
|------------|-----------|---|---|---|---|---|---|---|---|----|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 1 | 4 | 4 | 4 | 4 | 5 | 5 | 4 | 4 | 5 | 4 |
| 2 | 5 | 4 | 4 | 4 | 5 | 5 | 4 | 4 | 5 | 4 |
| 3 | 5 | 5 | 5 | 5 | 5 | 5 | 4 | 4 | 5 | 5 |
| 4 | 4 | 5 | 5 | 5 | 5 | 5 | 4 | 4 | 5 | 5 |
| 5 | 4 | 4 | 4 | 4 | 5 | 5 | 4 | 4 | 5 | 4 |
| 6 | 4 | 5 | 5 | 5 | 5 | 5 | 4 | 4 | 5 | 5 |
| 7 | 5 | 5 | 5 | 5 | 4 | 4 | 4 | 4 | 4 | 5 |
| 8 | 4 | 5 | 5 | 5 | 5 | 5 | 5 | 4 | 4 | 5 |
| 9 | 4 | 5 | 5 | 5 | 5 | 5 | 5 | 4 | 4 | 5 |
| 10 | 5 | 5 | 5 | 5 | 4 | 4 | 5 | 4 | 4 | 4 |
| 11 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 5 |
| 12 | 4 | 4 | 5 | 4 | 4 | 4 | 4 | 4 | 5 | 5 |
| 13 | 5 | 4 | 5 | 4 | 5 | 5 | 4 | 4 | 5 | 5 |
| 14 | 3 | 5 | 5 | 5 | 5 | 5 | 5 | 4 | 5 | 3 |
| 15 | 4 | 5 | 5 | 5 | 5 | 5 | 4 | 4 | 4 | 5 |
| 16 | 4 | 4 | 4 | 4 | 5 | 5 | 4 | 4 | 5 | 5 |
| 17 | 4 | 5 | 5 | 5 | 5 | 5 | 4 | 4 | 4 | 4 |
| 18 | 3 | 5 | 5 | 5 | 4 | 5 | 4 | 4 | 3 | 4 |

B. Perhitungan

Perhitungan reliabilitas instrument menggunakan metode *alpha cronbach*. Proses perhitungan menggunakan aplikasi SPSS 16.0. Hasil pengujian reliabilitas instrument menggunakan metode *alpha cronbach* yaitu:

| | soal1 | soal2 | soal3 | soal4 | soal5 | soal6 | soal7 | soal8 | soal9 | soal10 | soal11 | soal12 | soal13 | soal14 | soal15 | soal16 | soal17 | soal18 |
|----|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| 1 | 4.00 | 5.00 | 5.00 | 4.00 | 4.00 | 4.00 | 5.00 | 4.00 | 4.00 | 5.00 | 4.00 | 4.00 | 5.00 | 3.00 | 4.00 | 4.00 | 4.00 | 3.00 |
| 2 | 4.00 | 4.00 | 5.00 | 5.00 | 4.00 | 5.00 | 5.00 | 5.00 | 5.00 | 5.00 | 4.00 | 4.00 | 4.00 | 5.00 | 5.00 | 4.00 | 5.00 | 5.00 |
| 3 | 4.00 | 4.00 | 5.00 | 5.00 | 4.00 | 5.00 | 5.00 | 5.00 | 5.00 | 5.00 | 4.00 | 5.00 | 5.00 | 5.00 | 5.00 | 4.00 | 5.00 | 5.00 |
| 4 | 4.00 | 4.00 | 5.00 | 5.00 | 4.00 | 5.00 | 5.00 | 5.00 | 5.00 | 5.00 | 4.00 | 4.00 | 4.00 | 5.00 | 5.00 | 4.00 | 5.00 | 5.00 |
| 5 | 5.00 | 5.00 | 5.00 | 5.00 | 5.00 | 5.00 | 4.00 | 5.00 | 5.00 | 4.00 | 4.00 | 4.00 | 5.00 | 5.00 | 5.00 | 5.00 | 5.00 | 4.00 |
| 6 | 5.00 | 5.00 | 5.00 | 5.00 | 5.00 | 5.00 | 4.00 | 5.00 | 5.00 | 4.00 | 4.00 | 4.00 | 5.00 | 5.00 | 5.00 | 5.00 | 5.00 | 5.00 |
| 7 | 4.00 | 4.00 | 4.00 | 4.00 | 4.00 | 4.00 | 4.00 | 5.00 | 5.00 | 5.00 | 4.00 | 4.00 | 4.00 | 5.00 | 4.00 | 4.00 | 4.00 | 4.00 |
| 8 | 4.00 | 4.00 | 4.00 | 4.00 | 4.00 | 4.00 | 4.00 | 4.00 | 4.00 | 4.00 | 4.00 | 4.00 | 4.00 | 4.00 | 4.00 | 4.00 | 4.00 | 4.00 |
| 9 | 5.00 | 5.00 | 5.00 | 5.00 | 5.00 | 5.00 | 4.00 | 4.00 | 4.00 | 4.00 | 4.00 | 5.00 | 5.00 | 5.00 | 4.00 | 5.00 | 4.00 | 3.00 |
| 10 | 4.00 | 4.00 | 5.00 | 5.00 | 4.00 | 5.00 | 5.00 | 5.00 | 5.00 | 4.00 | 5.00 | 5.00 | 5.00 | 3.00 | 5.00 | 5.00 | 4.00 | 4.00 |

Gambar 1. Proses Penghitungan Reliabilitas Menggunakan Aplikasi SPSS

Reliability Statistics

| Cronbach's Alpha | N of Items |
|------------------|------------|
| .822 | 18 |

Gambar 2. Hasil Perhitungan Reliabilitas Instrumen

C. Kesimpulan

Berdasarkan hasil yang ditunjukkan oleh gambar 1 dan 2 diatas menunjukan bahwa reliabilitas instrument penelitian yang digunakan sebesar 0.822. Menurut Nunnaly dalam Jogiyanto(2008:141) nilai reliabilitas 0.50 sampai dengan 0.60 dianggap cukup. Berdasarkan hal tersebut dapat disimpulkan bahwa instrument penelitian yang digunakan peneliti lolos uji reliabilitas.

Lampiran 8 Surat Ijin Penelitian



KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
UNIVERSITAS NEGERI YOGYAKARTA
FAKULTAS TEKNIK

Alamat : Kampus Karangmalang, Yogyakarta, 55281

Telp. (0274) 586166 psu. 276.289.292 (0274) 586734 Fax. (0274) 586734

website : <http://ft.uny.ac.id> e-mail: ft@uny.ac.id ; teknik@uny.ac.id



Certificate No. QSC 00592

Nomor : 1782/H34/PL/2014

09 Juni 2014

Lamp. :

Hal : Ijin Penelitian

Yth.

- 1 . Gubernur DIY c.q. Ka. Biro Adm. Pembangunan Setda DIY
- 2 . Gubernur Provinsi DIY c.q. Ka. Bappeda Provinsi DIY
- 3 . Bupati Kabupaten Bantul c.q. Kepala Badan Pelayanan Terpadu Kabupaten Bantul
- 4 . Kepala Dinas Pendidikan, Pemuda , dan Olahraga Provinsi DIY
- 5 . Kepala Dinas Pendidikan, Pemuda , dan Olahraga Kabupaten Bantul
- 6 . Kepala SMK N 1 Bantul

Dalam rangka pelaksanaan Tugas Akhir Skripsi kami mohon dengan hormat bantuan Saudara memberikan ijin untuk melaksanakan penelitian dengan judul Pengembangan Aplikasi Android Kamus Command Line Sebagai Media Bantu Belajar Siswa SMK N 1 Bantul Jurusan Teknik Komputer dan Jaringan, bagi mahasiswa Fakultas Teknik Universitas Negeri Yogyakarta tersebut di bawah ini:

| No. | Nama | NIM | Jurusan | Lokasi |
|-----|--------------|-------------|-------------------------------|----------------|
| 1 | Afied Hutomo | 10520241008 | Pend. Teknik Informatika - S1 | SMK N 1 Bantul |

Dosen Pembimbing/Dosen Pengampu :

Nama : Dr. Eko Marpanaji, MT

NIP : 19670608 199303 1 001

Adapun pelaksanaan penelitian dilakukan mulai Juni s/d September 2014.

Demikian permohonan ini, atas bantuan dan kerjasama yang baik selama ini, kami mengucapkan terima kasih.

Bekas,
Wakil Dekan I
Dr. Simaryo Soenarto
NIP. 19580630 198601 1 001

Tembusan :
Ketua Jurusan



PEMERINTAH DAERAH DAERAH ISTIMEWA YOGYAKARTA
SEKRETARIAT DAERAH
Kompleks Kepatihan, Danurejan, Telepon (0274) 562811 - 562814 (Hunting)
YOGYAKARTA 55213

gubernur@pda.go.id

SURAT KETERANGAN / IJIN

070/REG/VI/249/6/2014

Membaca Surat : **WAKIL DEKAN I FAKULTAS TEKNIK** Nomor : **1782/H34/PL/2014**
Tanggal : **9 JUNI 2014** Penhal : **IJIN PENELITIAN/RISET**

- Mengingat :
1. Peraturan Pemerintah Nomor 41 Tahun 2006, tentang Perizinan bagi Perguruan Tinggi Asing, Lembaga Penelitian dan Pengembangan Asing, Badan Usaha Asing dan Orang Asing dalam melakukan Kegiatan Penelitian dan Pengembangan di Indonesia;
 2. Peraturan Menteri Dalam Negeri Nomor 20 Tahun 2011, tentang Pedoman Penelitian dan Pengembangan di Lingkungan Kementerian Dalam Negeri dan Pemerintah Daerah;
 3. Peraturan Gubernur Daerah Istimewa Yogyakarta Nomor 37 Tahun 2008, tentang Rincian Tugas dan Fungsi Satuan Organisasi di Lingkungan Sekretariat Daerah dan Sekretariat Dewan Perwakilan Rakyat Daerah;
 4. Peraturan Gubernur Daerah Istimewa Yogyakarta Nomor 18 Tahun 2009 tentang Pedoman Pelayanan Perizinan, Rekomendasi Pelaksanaan Survei, Penelitian, Pendataan, Pengembangan, Pengkajian, dan Studi Lapangan di Daerah Istimewa Yogyakarta.

DIJINKAN untuk melakukan kegiatan survei/penelitian/pendataan/pengembangan/pengkajian/studi lapangan kepada:

Nama : **AFIED HUTOMO** NIP/NIM : **10520241008**
Alamat : **FAKULTAS TEKNIK, PENDIDIKAN TEKNIK INFORMATIKA, UNIVERSITAS NEGERI YOGYAKARTA**
Judul : **PENGEMBANGAN APLIKASI ANDROID KAMUS COMMAND LINE SEBAGAI MEDIA BANTU BELAJAR SISWA SMK NEGERI 1 BANTUL JURUSAN TEKNIK KOMPUTER DAN JARINGAN**
Lokasi : **DINAS PENDIDIKAN, PEMUDA DAN OLAHRAGA DIY**
Waktu : **10 JUNI 2014 s.d 10 SEPTEMBER 2014**

Dengan Ketentuan

1. Menyediakan surat keterangan/ijin survei/penelitian/pendataan/pengembangan/pengkajian/studi lapangan *) dari Pemerintah Daerah DIY kepada Bupati/Walikota melalui institusi yang berwenang mengeluarkan ijin dimaksud;
2. Menyediakan soft copy hasil penelitiannya baik kepada Gubernur Daerah Istimewa Yogyakarta melalui Biro Administrasi Pembangunan Setda DIY dalam compact disk (CD) maupun mengunggah (upload) melalui website adbang.jogjapro.go.id dan menurahkan cetakan asli yang sudah disahkan dan dibubuhi cap institusi;
3. Ijin ini hanya dipergunakan untuk keperluan ilmiah, dan pemegang ijin wajib mentaati ketentuan yang berlaku di lokasi kegiatan;
4. Ijin penelitian dapat diperpanjang maksimal 2 (dua) kali dengan menunjukkan surat ini kembali sebelum berakhir waktunya setelah mengajukan perpanjangan melalui website adbang.jogjapro.go.id;
5. Ijin yang diberikan dapat dibatalkan sewaktu-waktu apabila pemegang ijin ini tidak memenuhi ketentuan yang berlaku.

Dikeluarkan di Yogyakarta
Pada tanggal **10 JUNI 2014**
A.n Sekretaris Daerah
Asisten Perekonomian dan Pembangunan



Hendar Sugeng, SH
NIP. 19590130-196503 2 003

Tembusan :



PEMERINTAH KABUPATEN BANTUL
BADAN PERENCANAAN PEMBANGUNAN DAERAH
(B A P P E D A)

Jln. Robert Wolter Monginsidi No. 1 Bantul 55711, Telp. 367533, Fax. (0274) 367796
Website: bappeda.bantulkab.go.id Webmail: bappeda@bantulkab.go.id

SURAT KETERANGAN/IZIN

Nomor : 070 / Reg / 2245 / S1 / 2014

Menunjuk Surat : Dari : Sekretariat Daerah DiY Nomor : 070/Reg/V/249/6/2014
Tanggal : 10 Juni 2014 Perihal : Ijin Penelitian

Mengingat : a. Peraturan Daerah Nomor 17 Tahun 2007 tentang Pembentukan Organisasi Lembaga Teknis Daerah Di Lingkungan Pemerintah Kabupaten Bantul sebagaimana telah diubah dengan Peraturan Daerah Kabupaten Bantul Nomor 16 Tahun 2009 tentang Perubahan Atas Peraturan Daerah Nomor 17 Tahun 2007 tentang Pembentukan Organisasi Lembaga Teknis Daerah Di Lingkungan Pemerintah Kabupaten Bantul;
b. Peraturan Gubernur Daerah Istimewa Yogyakarta Nomor 18 Tahun 2009 tentang Pedoman Pelayanan Perijinan, Rekomendasi Pelaksanaan Survei, Penelitian, Pengembangan, Pengkajian, dan Studi Lapangan di Daerah Istimewa Yogyakarta;
c. Peraturan Bupati Bantul Nomor 17 Tahun 2011 tentang Ijin Kuliah Kerja Nyata (KKN) dan Praktek Lapangan (PL) Perguruan Tinggi di Kabupaten Bantul.

Diizinkan kepada

Nama : **AFIED HUTOMO**
P. T / Alamat : **Fak Teknik ,Pendidikan Teknik Informatika UNY**
NIP/NIM/No. KTP : **10520241008**
Tema/Judul : **PENGEMBANGAN APLIKASI ANDROID KAMUS COMMAND LINE SEBAGAI MEDIA BANTU BELAJAR SISWA SMK NEGERI 1 BANTUL JURUSAN TEKNIK KOMPUTER DAN JARINGAN**
Kegiatan : **SMK NEGERI 1 BANTUL**
Lokasi : **11 Juni s.d 11 September 2014**
Waktu : **11 Juni s.d 11 September 2014**

Dengan ketentuan sebagai berikut :

1. Dalam melaksanakan kegiatan tersebut harus selalu berkoordinasi (menyampaikan maksud dan tujuan) dengan institusi Pemerintah Desa setempat serta dinas atau instansi terkait untuk mendapatkan petunjuk seperlunya;
2. Wajib menjaga ketertiban dan mematuhi peraturan perundangan yang berlaku;
3. Izin hanya digunakan untuk kegiatan sesuai izin yang diberikan;
4. Pemegang izin wajib melaporkan pelaksanaan kegiatan bentuk softcopy (CD) dan hardcopy kepada Pemerintah Kabupaten Bantul c.q Bappeda Kabupaten Bantul setelah selesai melaksanakan kegiatan;
5. Izin dapat dibatalkan sewaktu-waktu apabila tidak memenuhi ketentuan tersebut di atas;
6. Memenuhi ketentuan, etika dan norma yang berlaku di lokasi kegiatan; dan
7. Izin ini tidak boleh disalahgunakan untuk tujuan tertentu yang dapat mengganggu ketertiban umum dan kestabilan pemerintah.

Dikeluarkan di : B a n t u l
Pada tanggal : 11 Juni 2014

A.n. Kepala,
Kepala Bidang Data
Rencanan dan Pengembangan,
u.b. Kasubid. DSP

Ir. Edi Purwanto, M.Eng
NIP. 19607181997031004

Lampiran 9 Surat Keterangan Telah Melakukan Penelitian



PEMERINTAH KABUPATEN BANTUL
DINAS PENDIDIKAN MENENGAH DAN NON FORMAL
SMK NEGERI 1 BANTUL
Jl. Parangtritis Km. 11 Sabdodadi Bantul 55702 Telp. 367156



SURAT KETERANGAN

Nomor : 552/SMK.1/LL/2014

Yang bertanda tangan pada surat ini :

Nama : Ir. Retno Yuniar Dwi Aryani
NIP : 19610622 199303 2 005
Pangkat / Gol : Pembina/IVa
Jabatan : Guru Madya / Kepala Sekolah
Instansi : SMKN 1 Bantul

Menerangkan bahwa :

Nama : AFIED HUTOMO
NIM : 10520241008
Sem/Jurusan/Fak/Univ : - / Pen. Teknik Informatika / Fak. Teknik/ UNY

Benar-benar telah melaksanakan penelitian tentang " *Pengembangan Aplikasi Android Kamus Command Line Sebagai Media Bantu Belajar Siswa SMK Negeri 1 Bantul Jurusan Teknik Komputer Dan Jaringan*" pada tanggal 11 – 13 Juni 2014 guna Penyusunan Tugas Akhir.

Demikian surat keterangan ini dibuat untuk dapat digunakan sebagaimana mestinya.

Bantul, 13 Juni 2014

Kepala Sekolah,



Ir. Retno Yuniar Dwi Aryani

NIP. : 19610622 199303 2 005

Lampiran 10 Surat Keterangan Pembimbing Skripsi

**FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA
NOMOR : 242/ELK/Q-I/XI/2013
TENTANG
PENGANGKATAN, N PEMBIMBING TUGAS AKHIR SKRIPSI
BAGI MAHASISWA FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA**

**DEKAN FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA**

- Menimbang : 1. Bahwa sehubungan dengan telah dipenuhi syarat untuk penulisan Tugas Akhir Skripsi bagi mahasiswa Fakultas Teknik Universitas Negeri Yogyakarta, perlu diangkat pembimbing.
2. Bahwa untuk keperluan dimaksud perlu ditetapkan dengan Keputusan Dekan.
- Mengingat : 1. Undang-undang Nomor 20 tahun 2003.
2. Peraturan Pemerintah RI Nomor 60 tahun 1999.
3. Keputusan Presiden RI: a. Nomor 93 tahun 1999; b. 305/M tahun 1999.
4. Keputusan Menteri Pendidikan dan Kebudayaan RI: Nomor 274/O/1999.
5. Keputusan Mendiknas RI Nomor 003/O/2001.
6. Keputusan Rektor UNY Nomor : 1160/UN34/KP/2011

MEMUTUSKAN

Menetapkan

- Pertama : Mengangkat Pembimbing Tugas Akhir Skripsi bagi mahasiswa Fakultas Teknik Universitas Negeri Yogyakarta sebagai berikut :

| | |
|-------------------|---|
| Nama Pembimbing | : Dr. Eko Marpanaji |
| Bagi mahasiswa | : |
| Nama/No.Mahasiswa | : Afied Hutomo / 10520241008 |
| Jurusan/ Prodi | : Pendidikan Teknik Elektronika / Pendidikan Teknik Informatika |
| Judul Skripsi | : <i>Analisis Pengembangan Aplikasi Android Kamus CLI terhadap Pemahaman Kompetensi Administrasi Server Siswa Kelas XII Jurusan TKJ SMK Negeri 1 Bantul</i> |

- Kedua : Dosen pembimbing disertai tugas membimbing penulisan Tugas Akhir Skripsi sesuai dengan Pedoman Tugas Akhir Skripsi.

- Ketiga : Keputusan ini berlaku sejak ditetapkan

- Keempat : Segala sesuatu akan diubah dan dibetulkan sebagaimana mestinya apabila di kemudian hari ternyata terdapat kekeliruan dalam Keputusan ini.



Ditetapkan : di Yogyakarta
Pada tanggal : 12 November 2013
Dekan

Dr. Moch. Bruri Triyono
NIP. 19560216 198603 1 003